SAARLAND UNIVERSITY

DOCTORAL THESIS

Supporting Lay Users in Privacy Decisions When Sharing Sensitive Data

Author:
Frederic RABER

A dissertation submitted towards
the degree Doctor of Engineering (Dr.-Ing.)

of the

Faculty of Mathematics and Computer Science
of Saarland University

Saarbrücken, 2020

Dean of the Faculty:
Prof. Dr. Thomas Schuster

Chair of the Committee:
Prof. Dr. Bernt Schiele

Reporters:
First reviewer: Prof. Dr. Antonio Krüger
Second reviewer: Dr. Katharina Krombholz

Academic Assistant:
Dr. Florian Daiber

Day of Colloquium:
23.09.2020

Abstract

Especially after the recent privacy scandals in social networks, privacy is getting more and more important to users. Although most users claim to value privacy, their online behavior speaks differently: Most of the privacy settings in their online environment, like social networks, or location sharing services, remain untouched and are not adapted to their privacy needs. In this thesis, I will present an approach to tackle this problem, based on two different pillars. The first part focuses on assisting users in choosing their privacy settings, by using machine learning to derive the optimal set of privacy settings for the user. In contrast to other work, our approach uses context factors as well as *individual factors* to provide a *personalized* set of privacy settings. The second part consists of a set of intelligent user interfaces to assist the users throughout the complete privacy journey, from defining friend groups allow targeted information sharing; through user interfaces for selecting information recipients, to find possible errors or unusual settings, and to refine them; up to mechanisms to gather in-situ feedback on privacy incidents, and investigating how to use these to improve a user's privacy in the future. Our studies have shown that including tailoring the privacy settings significantly increases the correctness of the predicted privacy settings; whereas the user interfaces have been shown to significantly decrease the amount of errors, especially unwanted disclosures, that are made when sharing information.

Zusammenfassung

Insbesondere nach den jüngsten Datenschutzskandalen in sozialen Netzwerken wird der Datenschutz für Benutzer immer wichtiger. Obwohl die meisten Benutzer behaupten Wert auf Datenschutz zu legen, verhalten sie sich online allerdings völlig anders: Sie lassen die meisten Datenschutzeinstellungen der online genutzten Dienste, wie z. B. von sozialen Netzwerken oder Diensten zur Standortfreigabe, unberührt und passen sie nicht an ihre Datenschutzanforderungen an. In dieser Arbeit werde ich einen Ansatz zur Lösung dieses Problems vorstellen, der auf zwei verschiedenen Säulen basiert. Der erste Teil konzentriert sich darauf, Benutzer bei der Auswahl ihrer Datenschutzeinstellungen zu unterstützen, indem maschinelles Lernen verwendet wird, um die optimalen Datenschutzeinstellungen für den Benutzer abzuleiten. Im Gegensatz zu anderen Arbeiten verwendet unser Ansatz Kontextfaktoren sowie *individuelle Faktoren*, um *personalisierte* Datenschutzeinstellungen zu generieren. Der zweite Teil besteht aus einer Reihe intelligenter Benutzeroberflächen, die die Benutzer in verschiedene Datenschutzszenarien unterstützen. Dies beginnt bei einer Oberfläche zur Definition von Freundesgruppen, die im Anschluss genutzt werden können um einen gezielten Informationsaustausch zu ermöglichen, bspw. in sozialen Netzwerken; über Benutzeroberflächen um die Empfänger von privaten Daten auszuwählen oder mögliche Fehler oder ungewöhnliche Datenschutzeinstellungen zu finden und zu verfeinern; bis hin zu Mechanismen, um In-Situ-Feedback zu Datenschutzverletzungen zum Zeitpunkt ihrer Entstehung zu sammeln und zu untersuchen, wie diese verwendet werden können, um die Privatsphäreeinstellungen eines Benutzers anzupassen. Unsere Studien haben gezeigt, dass die Verwendung von individuellen Faktoren die Korrektheit der vorhergesagten Datenschutzeinstellungen erheblich erhöht. Es hat sich gezeigt, dass die Benutzeroberflächen die Anzahl der Fehler, insbesondere versehentliches Teilen von Daten, erheblich verringern.

Relevant publications

The work presented in this thesis, including figures and text fragments have in some cases appeared in the following publications. The chapters of this dissertation are partly based on these publications.

Full papers:

Frederic Raber and Antonio Krüger. "Towards Understanding the Infuence of Personality on Mobile App Permission Settings". In: *16th IFIP TC 13 International Conference on Human-Computer Interaction*. INTERACT. Springer, 2017.

F. Raber and A. Krüger. "Deriving Privacy Settings for Location Sharing: Are Context Factors Always the Best Choice?" In: *2018 IEEE Symposium on Privacy-Aware Computing (PAC)*. IEEE, Sept. 2018, pp. 86–94.

Frederic Raber, Christopher Schommer, and Antonio Krüger. "FriendGroupVR: Design Concepts Using Virtual Reality to Organize Social Network Friends". In: *17th IFIP TC 13 International Conference on Human-Computer Interaction*. INTERACT. Springer, Aug. 2019, pp. 654–658.

Frederic Raber, David Ziemann, and Antonio Krüger. "The 'Retailio' Privacy Wizard: Assisting Users with Privacy Settings for Intelligent Retail Stores". In: *3rd European Workshop on Usable Security*. Ed. by Charles Weir and Michelle Mazurek. EuroUSEC. Internet Society, 2018.

Currently under review:

Frederic Raber and Antonio Krüger. "Transferring Recommendations through Privacy User Models across Domains". In: *User Modeling and User-Adapted Interaction* (). Springer.

Short papers, late-breaking work and posters:

Frederic Raber and Antonio Krüger. "Applications for In-Situ Feedback on Social Network Notifications". In: *17th IFIP TC 13 International Conference on Human-Computer Interaction*. INTERACT. Springer, Aug. 2019, pp. 654–658.

Frederic Raber and Antonio Krüger. "OmniWedges: Improved Radar-Based Audience Selection for Social Networks". In: *17th IFIP TC 13 International Conference on Human-Computer Interaction*. INTERACT. Springer, Aug. 2019, pp. 654–658

Frederic Raber and Antonio Krüger. "Privacy Perceiver: Using Social Network Posts to Derive Users' Privacy Measures". In: *Adjunct Publication of the 26th Conference on User Modeling, Adaptation and Personalization*. UMAP '18. ACM, 2018, pp. 227–232

Frederic Raber and Nils Vossebein. "URetail: Privacy User Interfaces for Intelligent Retail Stores". In: *16th IFIP TC 13 International Conference on Human-Computer Interaction*. INTERACT. Springer, 2017, pp. 473–477

Frederic Raber et al. “Fine-grained Privacy Setting Prediction using a Privacy Attitude Questionnaire and Machine Learning”. In: *16th IFIP TC 13 International Conference on Human-Computer Interaction*. INTERACT. IFIP. Springer, 2017

Acknowledgements

The research of the last few years that allowed me to write this thesis would not have been possible without the aid of several people, whose support I would like to mention here.

First of all, I would like to thank my supervisor *Antonio Krüger*, who gave me the opportunity to return to this research area. He always supported my work since day one, helped me in shaping the dissertation topic, and was always available to give me feedback and guide me in the right direction, even if time was short.

I would also like to thank *Alexander De Luca* who supported me in the first days of my research on Usable Privacy, and *Katharina Krombholz* for giving me the opportunity to discuss my research with her, as well as the review of this thesis and many other bachelor's and master's theses.

My work also would not have been possible without the steadfast support of my colleagues at DFKI, who always provided a good working atmosphere and had an open ear for discussing my work.

Throughout my time at DFKI, I also had the opportunity to be the supervisor of many bachelor's and master's degree students, who performed a great deal of research that helped me to write this thesis. Special thanks also go to Margaret De Lap, who proofread every single word of this thesis and many of my other publications. I'm sure that their contribution had a significant impact on the quality of my work throughout the last few years.

Besides the professional support I received, I am very grateful for my parents, *Manfred* and *Marita*, who allowed me to develop my capabilities in the field of electronics, IT and computer science since I was a young child and supported me as much as they could. I would also like to mention my fiancée *Constanze*, who supported me and my work throughout the last few years. Although her expertise lies in another subject area, her ideas and my discussions with her often allowed me to solve problems or to improve my work.

Contents

Chapter 1

Introduction

In this chapter, we will give an introduction on the history of privacy, how privacy affects the digital world and the term "data privacy", and which problems and challenges occur within this context. Especially the development of privacy laws in the United States will be discussed in more detail, as most of the social network providers and internet companies like Google operate in this country. We will put a special emphasis on the topic of privacy in social media, where related work has shown the difference between perceived privacy and actual privacy, e.g. the perceived recipients of a data item (like a post or a photo) and the actual recipients differ significantly [34]. Based on the identified problems, we will discuss current approaches, as well as the problems that are still present. Research questions will be derived based on the identified problems that will be solved throughout the remainder of the thesis. As we will discuss later throughout this chapter, there are multiple touchpoints, e.g. multiple situations in the privacy journey of a user, where the approaches discussed in this thesis can assist the user in making her privacy decisions. At the end of this chapter, we will discuss these touchpoints and how our approaches can contribute to addressing the current problems that arise in those situations.

1.1 The history of privacy

The contents of this section are based on the book "The Death of Privacy: The Battle for Personal Privacy in the Courts, the Media, and Society" by Gini Graham Scott [289]. The notion of privacy has always been changing over time [289, pp. 3]. The term "privacy" was first mentioned in ancient times by Greek and Roman philosophers. At that time, people lived in close quarters in small dwellings consisting of only one room, or they lived together in a larger family compound, therefore people's lives usually were not very private. The term privacy had a negative connotation, persons demanding privacy were seen as deprived of something [289, p. 16]. Aristotle even claimed that "an individual who lived only a private life could not be fully human" [217]. The right to personal privacy requires the right to individualism, e.g. being able to choose whether to share aspects of one's personal life with the community and authorities or not. However, early cultures showed little or no concern for personal privacy or rights to individualism [35]. Slavery was a common practice in ancient Rome, and slaves were kept in barnlike accomodations like animals without any rights. Also for normal citizens, an unusual behavior was considered subject to the law and open to state scrutiny [35]. In contrast to that, the right to individualism and privacy against authority was an integral part of the Hebrew society. However, the right to privacy was still bound by numerous laws that told them what they could do or could not in their private life, such as what they were allowed to eat and whom they could marry [223]. This radically changed with the

introduction of the first democracy in Athens later, where people had the power to take part in the legislative process for the first time, allowing them to include their individual needs in the lawmaking process [232]. After the Roman Empire until the Middle Ages, the Christian church gained more and more power and thereby took over control over people's lives. Their religion required them to share their most private experiences, as they had to reveal any thoughts and actions that did not conform to the church's ideals during confession [156, p. 4]. However, the community evolved and later in the fifteenth century, a society of the privileged emerged in the West, where privacy was seen as a privilege rather than unsocial behavior. Only privileged people had the possibility to live in their own private house, pay for a better private education for their children, and be allowed to join private clubs, whereas common people were still living crowded together as in ancient times [342]. This notion of privacy as a kind of privilege was even reinforced in the sixteenth to the eighteenth century, where privacy in terms of a "private life" was seen as a key aspect for modern individuality and self-interest [223].

In the United States, early privacy laws were based on the English common law which denotes privacy as the right to be protected from "physical interference of life and property" [306, pp. 9-11]. People's home was seen as their private castle which should not be accessed without permission of the owner. The rights were later extended to cover the "spirital nature, feelings and intellect" of a person, eventually covering a "right to be let alone", including not only property but all kinds of possession, "intangible as well as tangible" [306, pp. 9-11]. After the Civil War, the urbanization of America and improvements in printing technology led to the rise of the media, which became affordable and popular among common people at that time, creating new privacy threats [289, p. 39]. Information could spread faster through media, and by the end of the nineteenth century, almost everyone who could read had access to a daily newspaper [289, p. 39]. At that time, popular journalism, presenting news in a sensational and colorful way, spreading gossip about well-known persons, was very popular. As a result, there was a growing concern about privacy threats [289, p. 40]. Oral gossip was limited to a few people and therefore not a big threat, but the new mass media were able to create an information circulation that had not existed before, especially for sensational and curious information [289, p. 40]. Whereas cameras were still too big and expensive at the beginning of the century, the privacy situation became worse as they became smaller and affordable at the end of the century, allowing photographs to be published in media as well. Also, advertisers used photographs of people, sometimes without their consent, and, even worse, sometimes suggested they would endorse their products [289, p. 40]. The result was a significant rise in the number of court cases against privacy violations, for example where a photo studio was using a picture of a person to sell Christmas cards without even notifying the person [289, p. 41]. The increasing number of privacy court cases led two lawyers named Samuel Warren and Louis Brandeis to work on an article about privacy for the Harvard Law Review in 1890. They pointed out that informational privacy was not part of the United States law at that time, and proposed that such laws should established in the future as a countermeasure to the privacy threats introduced by the media, based on the "right to be let alone" [335]. Brandeis, who became a judge in the powerful Supreme Court later, strengthened the privacy law in the United States based on this article [289, pp. 42]. At the same time, privacy against the government also became an important discussion point with an increasing number of court cases, as the state used to search people's homes and used wiretapping and eavesdropping techniques on telephone devices that became popular in the 1920s [289, pp. 44].

As we can see, the way in which privacy is protected or violated changes when new technology is introduced. Every technology that allows the spread or to gathering of information in a new way offers new ways of breaching privacy which are not yet covered by law, leading to an increased privacy threat and thus an increased number of court cases [289, pp. 56]. In the nineteenth century, the evolution of mass media posed the biggest threat to privacy, followed by an increasing use of surveillance techniques by the state in the twentieth century [289, pp. 44]. At the end of the twentieth century, a new information technology called the internet arose, making it again easier to spread, store and access information on demand, posing completely new privacy threats[1], which will be discussed in the next section.

1.2 Privacy and the internet

Privacy concerns regarding online data sharing have been articulated since the beginning of the internet, when computers were connected and shared around the world [84]. The term "internet privacy" thereby involves the storing, repurposing, sharing with third parties, and displaying of information about a person's private life, meaning all information relating to an identified or identifiable individual [190]. That means that information privacy entails either personally identifying information (PII), e.g. information which allows identifying a person with a high probability without mentioning the name, for example through the age and the physicial address of a person, or GPS tracking data [193], as well as non-PII information such as the visitor's behavior on a website. The internet allows information to be spread even faster than in the age of growing media, especially because social network sites also allow users to create and publish their own content [330]. Although lawmakers have tried to limit the unauthorized sharing of sensitive data, there are many threats to privacy, leading to a general opinion that "privacy is dead" [255]. Throughout this section, we will describe the privacy threats on the internet and especially in social media.

One of the oldest and most often discussed privacy threats in the internet is HTTP cookies [250]. Cookies are typically used to store state information required on complex websites, for example for storing the contents of the shopping cart on an online shopping website. However, cookies can also be used for tracking a user's behavior throughout the internet, for example which websites he has visited, and, based on that, which topics and products he is interested in. These *tracking cookies* make it possible to build a profile of a user's behavior and interests and can also be used in computer forensics, and therefore pose a privacy concern that has prompted lawmakers in Europe and the US to take action in 2011, resulting in laws that forbid tracking users throughout the internet without their knowledge and require compliance with users decisions not to be tracked[2]. Whereas most users were not aware of tracking cookies at the rise of the internet in the 90's, later studies in 2009 have shown that at that time, about 58% have deleted a cookie at least once, and 39% did it every month[3]. Therefore, developers were searching for ways to circumvent this problem, so they would still be able to track the user even if she completely disabled HTTP cookies. Adobe Flash also allowed developers to create cookies, which are not

[1] http://archive.nytimes.com/www.nytimes.com/2010/07/25/magazine/25privacy-t2.html (last accessed: 2020-03-09)

[2] https://www.theverge.com/2019/5/20/18632363/sen-hawley-do-not-track-targeted-ads-duckduckgo (last accessed: 2020-03-09)

[3] https://www.comscore.com/Insights/Press-Releases/2007/04/comScore-Cookie-Deletion-Report?cs_edgescape_cc=DE (last accessed: 2020-03-09)

affected by the aforementioned countermeasures against HTTP cookies that users began to carry out. In 2009, Flash cookies, a.k.a. local shared objects, were shown to be the most popular technique for storing data on the 100 most visited web sites [307]. Two years later, a study on social media discovered that of the top 100 websites, 31 were using both HTTP and Flash cookies [252]. As browsers became able to automatically detect and remove Flash cookies as well, website programmers began to develop a mechanism known as *zombie cookies*, which store the tracking data at multiple destinations, with the ability to recover missing copies if deleted by the user [2]. Its most famous implementation, called *Evercookie*, by Samy Kamkar[4] is able to store the information in more than ten types of storage mechanisms, for example Flash cookies, various HTML5 storage mechanisms, using caching mechanisms, etc. If only one of the copies is deleted, Evercookie uses one of the still existing storage mechanisms to recover the deleted cookie. Evercookies are still used as one of the major techniques for behavior-based targeted advertising [43]. Another technique that makes it possible to reidentify and track a user is device fingerprinting [161]. Instead of storing the information on the device, device fingerprinting uses the properties of the device, like hardware IDs, the MAC address, the TCP/IP configuration or the OS fingerprint. By that means, it is possible to discriminate between devices, although it is often, especially on mobile devices, not always possible to uniquely identify a device using fingerprinting [161]. But in contrast to cookies, it is hard for a user to prevent fingerprinting, as sending wrong hardware information can affect the layout of the website in a negative way and also slow down the speed of the web browser. Similar to cookies, device fingerprinting is mostly used by advertisers[5].

Another topic regarding privacy in the internet is the sharing of photos, and especially tagging of persons appearing on a photo. Whereas the audience of a shared image has typically been limited to friends and relatives before the rise of the internet [60], the audience for snapshots is nowadays significantly larger. Photos are not taken solely for home usage anymore; rather, they are created for public appeal, and are often posted on social media websites accessible to the public [203]. However, users often underestimate the audience of photos when posted online, leading to a privacy risk [34]. Especially problematic in social networks is the procedure of *tagging*, which allows the original poster of the photo to name and link the persons visible on the photo, even if they never agreed to being tagged. By that means, a user can upload photos including a person and even displaying the person's identity without any consent of the tagged person. As a Harvard law review article has shown, there is not much a person can do against being unwillingly tagged [164]. It takes some time for the person to find out about being visible and tagged in a published photo, and further time for the social network provider to take the photo down after a complaint from the user. During this timespan, the photo displaying the person in a way that could violate them personally can be seen, shared and distributed, without the possibility of ever deleting all of its copies [164]. Using the published pictures, it is even possible to re-identify a person online and also offline, allowing the creation of augmented reality apps that recognize persons in real life and display their publicly available information to a user [4]. Also, Google Street View has been seen critically in terms of privacy, as the pictures may show an individual's involvement in particular activities [99].

Early web search engines in the 2000s like Yahoo and AOL Search already had the ability to track a user's searches, for example through the IP address or other

[4] https://samy.pl/evercookie/ (last accessed: 2020-03-09)

[5] https://blogs.wsj.com/digits/2010/11/30/how-to-prevent-device-fingerprinting/ (last accessed: 2020-03-09)

identification meachanisms discussed above. Although the user's identity was not known, those search engines could already find out a lot about the individual's interests and current needs [6]. A grocery chain named *Target* was even able to determine a young woman's pregnacy before her own father, sending out coupons for baby supplies to their common home address [7]. After 2009, Google launched their personalized search engine, which uses the user profiles created by search requests and visited websites to offer the user individualized search results that should better match her needs [17]. Besides individualized search results, Google also uses the profiles for personalized advertisements [356]. Studies have shown that personalization of pre-purchase advertisements, i.e. advertisements that are shown to the user when he is about to look for a product, as well as post-purchase advertisements (i.e. ads that are shown after the user bought a product) for alternative products significantly increases the click rate on the displayed ads [356]. Since 2012, Google is using its account system to track users and their interests over multiple of the user's devices, like different computers, laptops and also smartphones [81]. Unfortunately, users are not given any chance to opt out from the data collection or targeted advertising; the only way to opt out is to delete all Google accounts, also losing access to its social network Google+ and the Play Store, which is the central way of installing and maintaining apps on an Android smartphone [8]. Other large internet companies like Amazon, Apple, Facebook or Spotify are also storing usage profiles of their visitors. Analyzing these massive data sets, also known by the term *Big Data Analysis*, allows the companies to infer detailed psycho-demographic profiles of their visitor without their notice [192].

Finally, internet service providers (ISPs) also have the ability to track the user's visited websites and, if the connection is not encrypted, also the actual content of the communication [138]. The privacy policies usualy state that such data is not analyzed; however, federal organisations can usually require an ISP to give them access to this information, in some countries like the United states even without a warrant. Mobile phone providers have even introduced additional header information into the HTTP header (a.k.a. *header enrichment*), which allows them to identify and track their users, which is also broadly used for the purpose of targeted advertising [329]. In the next section, we will first define the terms privacy and lay users as they are used nowadays, and then discuss existing privacy concerns of lay users in social media websites and location sharing, which is the major domain that we want to address within this thesis.

1.3 The latest notion of "privacy" and "lay users"

1.3.1 Privacy and sensitive data

As stated in the beginning of this chapter, the term "privacy" has been described formally first by Warren and Brandeis as "the right to be left alone" [130, pp. 15-17]. However, there has been a lot of discussion over the actual meaning of being "left alone". One of the most recognized publications about this term by Daniel Solove defines privacy as the right to seclude from other persons and their surveillance,

[6] https://www.cnet.com/news/aols-disturbing-glimpse-into-users-lives/ (last accessed: 2020-03-09)

[7] https://www.forbes.com/sites/kashmirhill/2012/02/16/how-target-figured-out-a-teen-girl-was-pregnant-before-her-father-did/ (last accessed: 2020-03-09)

[8] https://www.washingtonpost.com/business/economy/google-privacy-policy-is-subject-of-backlash/2012/01/25/gIQAzwZCRQ_story.html (last accessed: 2020-03-09)

and to be immune to their scrutiny in the person's private space, for example in his home environment [305, pp. 15-17]. Besides this right to be left alone, the term of privacy is defined by several other so-called *aspects of privacy* in literature, which we will explain in the remainder of this subsection.

One of them is the *limitation of access* to private data, meaning that a person should be able to be part of a society without other individual organizations or persons collecting private data about them [305, p. 19], meaning that privacy is the ability of a person to limit access to her own personal information or "the condition of being protected from unwanted access by others—either physical access, personal information, or attention" [44, p. 10-11]. Similarly, another privacy aspect is a person's *control over information*, which is defined by Alan Westin as follows: "privacy is the claim of individuals, groups, or institutions to determine for themselves when, how, and to what extent information about them is communicated to others" [119].

Alan Westin has also defined four *states of privacy* [119] namely *solitude*, meaning a "physical separation from others"; *intimacy* as a "close, relaxed, and frank relationship between two or more individuals" [119], *anonymity*; meaning the "desire of individuals for times of 'public privacy." [119] and lastly *reserve* denoting the "creation of a psychological barrier against unwanted intrusion" [119] of a person, which have to be respected by others by restricting communication of information concerning the person. Another definition of privacy and another *privacy aspect* is *secrecy*, which has been defined by Richard Posner as the right to hide personal information from others that could be used to their disadvantage [191, p. 271], also taking into account that the unwanted disclosure of information can also lead to negative consequences for a person, and that the right to privacy should protect a person from those negative consequences.

Privacy can also be seen as a necessary precondition for a person to build her own personality and identity, whereas the lack of privacy can hinder this development [275]. Through privacy seen as a "social ritual", children are thought that they have moral ownership over their body, which also entails the control over who is allowed to meet them and when [275]. Others defined privacy as a recognition of people as individuals that can make choices *autonomously* about themselves; therefore, privacy means the ability to choose (regarding which private information is shared) [32]. By that means, privacy can also be understood as a requirement for person to build a *self-identity*. Privacy barriers are elemental in this process, as they allow to characterize the boundaries and limits of the character and thereby help to shape the character of a person [10]. This also includes the ability to decide when to have contact with other persons and their perceived image of a person's self, which enables a person to control which external factors establish the self and thereby shape the character and the self-image [10].

Similarly, privacy is also described as an essential requirement for persons to strengthen or intimate relationships with other humans [305, p. 35]. As a part of relationship with another, it is necessary to share personal information with a friend and also to receive personal information from that person, while some information is not shared. On the other hand, social norms also require withholding some of the private information that a person received within his relationship, whereas other information might be shared [305, p. 35].

In this thesis, privacy and sensitive data are meant as the possibility of the user to have an overview of the data that is collected, as well as the control over which data is collected, and with whom it is shared. With the term of privacy, we therefore mean *data privacy*, e.g. control and autonomous decision on the collection, processing and sharing of personal digital data in general, shown as an example for data that is shared in a social media context, as well as data acquired and processed by mobile phone apps and intelligent retail stores.

1.3.2 Lay user

Although the term of a lay person is commonly used, not many attempts have been made to define the term of a "lay user" or "novice user" [62]. Checkoway defined lay persons as persons without some kind of financial vested interest in decision-making outcomes of a domain [63]. Lundval et al. discriminated lay and professional users by the definition of their goals in the domain: A professional user is a user who has a well-identified goal for his/her activities, acting within the formal part of the economy [216]. Alternatively, lay persons can also be defined as persons without an abstract body of knowledge that can be applied in the domain in question [14]. Although lay users have little or no domain experience at the beginning, they acquire specialized domain knowledge when involved in a domain-specific task, and can furthermore bring different types of knowledge from other areas that can be useful [30]. Lay users are mostly defined by other terms like "non-professionals", "non-experts" or "amateurs", which means they are defined more by abilities they do not have, rather than the abilities they do have [157]. Lay persons are those "who have not gone through the training or socialization into the particular profession under discussion" [157]. Lay persons are therefore persons that have the "ordinary" social norms, knowledge, and skills of the society. In contrast to that, professionals acquire new social norms and skills during their training and professional practice [343]. The skills and social norms acquired throughout this process can be traced through reading professional publications and talks that are typically published in the domain. However, the skills and social norms cannot be defined so easily.

This definition of a lay person has several negative aspects: First, this definition distinguishes only lay users and professional users regarding a specific domain called the index domain. However, a part of the population that is called a "lay user" by this definition could also have *some* prior experience in a similar field. A Java programmer would be called a lay user in Python programming according to the definition. However, her experiences in the Java programming domain will also be of use in the Python programming domain; she therefore cannot be compared to another lay user without any coding experience. Also, users that are not proficient in a similar domain may bring other experiences with them that are more or less helpful; for example, if they are using a computer on a daily basis in their life. Lastly, some studies and decisions are made with expert members only, which eases the decision process [287]. Still, studies have shown that lay people can also contribute with their knowledge in such scenarios, and should therefore not be excluded a priori [157].

Lay and expert users both have their own characteristics which a developer has to account for during the software development process. Especially expert users have their expectation for how a product or user interface will work, based on the experiences they have had with similar products in the past [176]. Furthermore, efficiency and effectiveness plays only a minor role for lay users, but pleasure and

early success are particularly important for them, as they motivate users toward further exploration of functionality and interaction [54]. Professional users usually understand a product because of their prior knowledge and similar products they have used before. However, this is not possible for lay and novice users; therefore, guess-ability is an important aspect, so they can explore the software on their own without prior technical knowledge [142].

Professional users typically have good ideas for overcoming limitations of the software [340] and respond to unexpected behavior of the software better than lay users [9]. Lay users, on the other hand, need support to overcome such problems and need to be pointed to workarounds for the software limitations [142]. Lay users also typically do not understand the specific terminology of the domain. Studies have found that there is a mismatch in information representation used by lay users compared to professional users [304].

Professional users also usually use the products in their work environment, and therefore often also have concerns about the legal consequences of their actions. Therefore, professional users tend to follow more closely the rules and instructions given by software developers. Lay users, in contrast, do not have to fear legal consequences, or consequences for their job position. Furthermore, lay users use the software typically at home or in an uncontrolled environment, whereas professional users are usually using the software in a public controlled environment [97].

Lastly, when a professional user wants to switch the software she uses, she usually wants to take advantage of her prior knowledge with the software and therefore will most likely choose a (software) product that has changed only a little compared to the product that she used before. Radically changing to a completely new product with a completely different design and functionalities would need a professional user to start again from scratch and get used to the new software, wasting accumulated experience [340]. Hence, lay users that have no prior knowledge about any product in the domain can be motivated more easily to choose new designs that are different from the current standard, and can profit from new design concepts that have not been tested before [80].

Based on the findings of related work, we designed our user interfaces using interface metaphors like the radar metaphor, which are currently *not used in commercial applications like social media websites or smartphone operating systems*, but which have been shown to perform better compared to those traditional interfaces according to user studies. Furthermore, the focus in our UI development lies on the side of *enjoyment and motivation* of the user, whereas pragmatic capabilities play only a minor role. According to the literature discussed above, these two decisions should make our user interfaces attractive especially for the target group of this thesis, namely the lay users.

1.4 Key problems in social media privacy

According to a study from 2016, about 73% of all adults in the U.S. are using social network sites today [259]. Privacy concerns are inherent to social media platforms, as "by design, social media technologies contain mechanisms for control and access to personal information, as the sharing of user-generated content is central to their function.", meaning social network companies need private data to be made publicly available in order to operate [259]. Once shared with the public, it is hard to

[9] `https://www.fda.gov/media/838888/download` (last accessed: 2020-03-09)

unshare social media content, as other users can forward the information to an audience different from what the original poster expected, or download the photos or create screenshots to store the content permanently, without the ability of the original poster to delete it. On the one hand, sharing information online can therefore create privacy concerns; on the other hand, it allows users to connect to people all around the world, stay in touch with friends gone abroad, or discuss special interests in groups independent of their location, religion, and race [337]. Therefore just staying away from social networks, or not actively engaging in social media, is also not a solution, as it leads to a reduced amount of social ties and a decreased social capital [98]. Whereas users had few privacy concerns in the beginning of social media [355], things have changed over the last years: In 2013, 60% of teenage Facebook profiles were already private [228].

Although social network platforms offer privacy settings which allow users to narrow down the audience of a post to single individuals, the privacy settings that users apply often differ from their desired privacy settings [27]. This effect called the *privacy paradox* can be found often in the domain of social networks, and might be caused by the *third person bias*, meaning that the people are aware of the privacy risks, but think that the risks do not apply to themselves as individuals [175]. Others speculate that users lack the technical knowledge needed to transfer their privacy desires into actual privacy settings, leading to the fact that users barely touch the privacy settings and thereby run into the privacy paradox [218]. Often, the risk of unintentionally spreading information is perceived as small compared to the appeal that a user can achieve when sharing private information on the social web exclusively for friends or followers [87].

As studies in the past have shown, users generally trust social network providers, although it is known that they use the data for advertisements [95]. Nevertheless, there are many other ways the data in a social network can be retrieved and misused by third parties without the support of the social network provider: Social network *apps* often request access to a user's personal profile to offer a personalized experience. Most of the popular apps on Facebook, for example Farmville and Quiz Planet, do so - not only for a personalized experience, but mainly as a part of their business model, which includes selling private data to advertisers and tracking companies [10]. It has been shown that the learning app *Take With Me Learning* is recording students' personal information like name, school, email and age and selling it to advertising companies without the consent of their users [248], thereby violating the Children's Online Privacy Protection Act (COPPA). Only the most recent Cambridge Analytica privacy scandal, where the data of participants of a personality test app as well as the data of the participant's friends was used to sway people's votes for the presidential election, came to the public and decreased the users' trust in the Facebook social network [135]. Social network sites usually offer an API for their applications to comfortably retrieve personal data from the app users, and are therefore also often used by researchers, in order to collect anonymous data samples [211]. It is also possible for hackers to gather access to private profiles by sending out friendship requests from a fake user. A study has shown that of 250,000 friendship requests from an unknown user, 75,000 were accepted, thus giving access to the private profile of the user which is visible for friends only per default [139].

Based on the gathered data, either illegal or legal, it is possible to perform several attacks: The hacker can use the gathered information to create a fake profile, which

[10] https://gawker.com/5666325/how-to-stop-facebook-from-sharing-your-information-with-third-parties (last accessed: 2020-03-09)

can be used for cyberbullying and stalking. It has also been shown to be possible to derive social security numbers out of the private data inside a social network [6], which is used as an authorization for several legal acts. One out of five employers search the social networks before hiring a canidate, in order to acquire negative information about the candidate [11]. Employers are especially concerned about alcohol and drug abuse, and pictures of excessive party situations in social media can be read as a tendency towards such problems [11]. Also, law enforcement and secret services have successfully been using the social networks to prevent criminal acts and to track down suspects [12]. Since 2017, the U.S. Department of Homeland Security (DHS) has been screening the social network profiles of immigrants arriving in the U.S. [13]. Having controversial content or contacts in one's social media profile can therefore also lead to the rejection of a visa for the United States.

Having these possible consequences in mind, it is clear that it is important for a social network user to narrow down their audience to the smallest possible subset, while still keeping in touch with friends, keeping the social ties stable. Unfortunately, this task is very hard for a user, as the next section will show.

1.5 Current approaches

Social network providers have already tried to make it easier for their users to perform narrowcasting, e.g. to narrow down the audience as much as possible to prevent unwanted data disclosure: On Facebook and Google+, *friend lists* and *circles* were introduced, which allow users to assign their friends to different groups, corresponding to the social circle they belong to (for example workmates, family, or friends from a football club). However, assigning all friends to the correct friend lists is a very time-consuming task [247], as a social network user on Facebook, for example, has on average 388 friends [14] up to 5000, the highest number of friends possible. Therefore, social network providers came up with the idea of *automated friend lists*, which are inferred automatically, for example using profile information like the home address, school attended, or the employer. However, those friend lists rarely reflect the actual social groups and thereby the friend lists a user would have created manually [230]. Therefore, friend lists are created very rarely [167], and only about 17% of all posts are shared using custom privacy settings, involving friend lists or excluding/including single persons [230]. Research has tried many different approaches using community detection [40] or machine learning based on the friends' profile data, activity logs or the friendship graph [230]; however no solution that is accepted by users has been found so far [230].

Although friend lists already make it easier to define the correct audience for a data item, the user still has to manually select the right groups and possibly include or exclude single persons. Research has therefore worked on *recommender systems* that propose privacy settings to the user. These recommender systems are either based on simple *rule-based* approaches [58] or on *machine learning* algorithms. Rule-based recommenders decide on the disclosure of a data item based on a hand-written rule set that was created by the developer, the user, or both. As an input for the decision, rule-based systems use **context factors**, like the topic of a post, the distance in

[11] https://www.computerworld.com/article/2532900/one-in-five-employers-uses-social-networks-in-hiring-process.html (last accessed: 2020-03-09)

[12] http://www.nbcnews.com/id/35890739 (last accessed: 2020-03-09)

[13] https://www.nytimes.com/2017/09/28/us/politics/immigrants-social-media-trump.html (last accessed: 2020-03-09)

[14] https://www.brandwatch.com/blog/facebook-statistics/ (last accessed: 2020-03-09)

the friend graph between original poster and recipient, or the occasion when a location is shared [58]. Rule-based systems have the advantage that it is easy to find out why the system recommended a given privacy setting by reading the rules. Some rule-based recommenders even allow one to specify the privacy rules in natural language [49], allowing lay users to refine and adapt the privacy settings. Machine learning-based systems, on the other hand, do not rely on a set of rules. Instead, machine learning systems are trained with a set of training data, which includes the input variables (for example context factors) together with the corresponding correct privacy settings the machine learning system has to predict later. After processing the training data set, the machine learning system is able to predict the privacy settings for given input variables. Also, some machine learning based recommenders use context factors like the topic of the post as an input for the prediction [300], whereas others try to use privacy settings the user already chose in the past [31], or ask users for explicit feedback on a sample of privacy decisions [296, 295] as an input for the training phase. Another possibility for recommending privacy settings is clustering the training set based on the personality of the users into a small finite set of clusters [128]. A new user is then assigned to one of the clusters based on her personality, and given the average privacy settings of that cluster as a recommendation. However, assigning a user to a cluster based on personality measures usually requires the user to fill out a personality questionnaire, introducing an additional user burden. Research has found ways to derive personality measures from the writing style of user-written text [104], which can be used as a solution for the problem. Interestingly, personality measures like the big five personal inventory and especially privacy measures have not yet been used as an input for a machine learning-based prediction, although they could be used as additional measures together with context factors to further increase the precision of the recommendation. The derivation of privacy measures from user-written text has also not been examined so far.

In some cases, neither a user model nor previous privacy decisions are available as an input for the recommendation, nor does the user have social network posts or other written text that can be used for deriving the user model. However, research has already succesfully used *cross-domain recommender systems* for solving such kinds of problems. If no data is available for a domain, cross-domain recommenders use the user model from another domain where more is known about the user, and try to transfer it to the domain in question. Although the recommendation precision is significantly lower compared to single-domain recommender systems [282], it is sometimes the only option to generate personalized recommendations when no or only a little domain data is available [282]. This technique has already been succesfully implementend in research as well as in commercial applications, for example for recommending books based on movies watched [345], or for recommending music that matches the environment at the current location [178]. The usage of cross-domain recommender systems to recommend privacy settings based on the privacy settings in other domains has not been part of research so far.

Apart from recommending privacy settings, researchers examined how improving user interfaces can help users in detecting errors and adapting their privacy settings. One often used approach is to try to reduce the complexity of privacy settings or privacy notifications to a minimum, so that users can directly recognize what the problem is, and how they could solve it. An example for this is the *privacy bird* [79], which pops up everytime the user visits a website that does not match their privacy preferences, so they can decide whether to stay or to leave. Based on the color of the displayed bird icon and the message contained, users can directly see whether the

problem is that no privacy policy is given by the website, and whether only embedded content or the whole webpage violates the desired privacy settings. Also, the *privacy nutrition label* by Kelley et al. allows users to easily compare two websites regarding their privacy specifications using a label similar to the well-known nutrition labels printed on retail products [182]. Although such UIs allow for a quick overview on the most important facts, they lose a certain degree of detail which might also contain important information.

A second approach is used by consequence based privacy UIs, which try to engage users in thinking over and adapting their privacy settings by highlighting possible consequences of their decision. Such a system has been implemented by Wang et al. for Facebook posts [333], where users are shown the profile pictures of some of the recipients after the user has pressed the send button, giving them three more seconds to cancel the final publication of the post. Similarly, other researchers published an approach that displays the excessive access rates of smartphone apps to private information like the current location [9]. Studies in both aforementioned publications have shown that the UIs significantly engaged the users in thinking over and also adapting their privacy settings.

Finally, research has also tried to improve the user interfaces for selecting the audience for a data item, and for adapting the privacy settings manually. Regarding the former, Kauer et al. proposed a UI for selecting the audience of a social network post based on interpersonal distance [180], e.g. the tie strength between the original poster and her friends, to align the recipients in the UI, and to allow the user to select the recipients up to a given tie strength, so that friends who are not close enough can be excluded from an intimate post. Their UI aligned the profile pictures from left to right and offered a slider that could be moved from the left to the right, selecting the audience for the post up to that point. Unfortunately, the UI was able to display only a limited number of friends due to the space needed by the profile pictures, making it hard to use for a real social network profile with hundreds of friends. A similar approach using a circular design to arrange the friends has also been proposed for selecting the recipients of an email [105].

Other studies in the past have shown that *radar interfaces* provide a better overview and usability in many cases, among others for defining the parameters for a music recommendation system [227], but also for specifying privacy settings, for example for participatory sensing applications [69]. According to these studies, the radar interface does not only allow the user to get a better and quicker overview on the UI, but also actively engages the user in adapting the privacy settings, as he can detect potentially critical settings better [67] than with a conventional interface containing only an endless list of on/off options. Unfortuantely, radar interfaces also have a limited amount of space, making it hard to display a large number of potential recipients inside a social network. Furthermore, there exist multiple parties that are interested in a specific type of sensitive information. The items a customer bought during a shopping trip, for example, is of interest for the retailer and for third parties like marketing agencies, as well as friends and family. Currently, radar interfaces do not support the display of privacy settings for multiple parties at once.

Lastly, there are several domains where in-situ feedback is used to capture users' problems and negative experiences when using the software, for example in social networks or mobile phone apps [114]. In-situ feedback has been proven to be very effective to bridge the gap between developer and user, and allows them to interact with each other, for example through error reports that can be sent when a user has experienced an application crash, also known as *remote evaluation* [313]. Remote evaluation is integrated into several products like Microsoft Windows and the

Ubuntu operating system. There are also solutions for in-situ feedback for home IT systems [155], manual assembly workplaces [123], mobile phone applications [290] and many other domains. However, the capture of privacy violations using in-situ feedback has not been part of research so far. In the next section, we will summarize the drawbacks of the aforementioned approaches.

1.6 Problem statement

As stated above, several privacy recommender systems are based on earlier privacy settings of the user. But often, for example when the user has just created a new social nertwork account, no information on prvious privacy decisions is available. This is also known as the *cold start problem* [222]. Some of the recommender systems actively ask for privacy decisions. However, answering these requests correctly requires the user to have technical knowledge about the consequences and privacy threats arising from the decisions, and produces an additional user burden. Furthermore, such systems suffer from the privacy paradox, meaning that users' privacy decisions often differ from their actual privacy desires, and thereby so do recommender systems which are based on users' past privacy decisions. Other privacy recommenders are based on context factors as an input for the recommendation. A third and last kind of approach clusters the data according to the user's personality, where the number of clusters is typically very low. Each cluster is assigned an averaged privacy policy of its members, so that a new user is first matched to the best cluster according to her personality, and is then given the cluster's averaged privacy policy as a recommendation. A recommender that is prone to the cold-start problem offering a regression-based recommendation using a combination of personality measures and privacy measures (later denoted as *individual measures*), which offers each user an individualized privacy setting rather than recommending all users in the same personality cluster the same privacy setting, has not been part of research so far.

Cross-domain *privacy* recommenders, e.g. recommender systems that recommend *privacy settings* for a domain using the privacy settings from other domains, have not been examined in the past. In particular which domains are suitable inputs for the prediction of privacy settings of another domain is completely unknown at the moment. Furthermore, the detail level of the input that produces the best results is unclear, for example whether a general privacy level of a domain leads to a lower prediction precision than having detailed privacy levels depending on context factors like the occasion and groups of recipients. Also, *which* detailed privacy settings should be used for the prediction and which prediction precision can be expected is not clear so far.

Studies have found that the social group of the requestor of an information item is an important context factor when deciding whether to allow or deny access [246, 321, 74]. Some privacy domains, for example social networks, therefore allow users to group their friends into groups, so these can be used when defining the privacy settings for a new post. However, as mentioned above, creating such groups introduces an additional burden, leading to the fact that this option is usually not used. Automatically deriving friend lists has also not led to satisfying solutions so far. A gamified approach examining how a grouping task can be designed to be more enjoyable and thereby motivating for users doing the task has not been part of research so far.

Recommender systems based on the user's personality usually require a user to fill out a privacy questionnaire, which can take up to 10-25 minutes depending on the questionnaire, introducing an additional user burden. Research in the past has shown that personality measures can also be derived from the user's written text, for example her email communication, blog entries, social network posts or YouTube comments [104]. So far, the derivation of privacy measures through written text, which will most likely also have a large influence on the privacy decision, has not been examined in research.

Recommender systems are never 100% accurate, but even one incorrectly set privacy setting can lead to an unwanted data disclosure and thereby to user frustration. Therefore, in our opinion, privacy recommenders should always be accompanied by other systems that help users to review and correct the recommendations easily. As stated before, user interfaces involving a radar design have been shown to make a big step towards a better overview on the current privacy state, thereby also motivating users in actively engaging in their privacy settings. However, the space inside a radar interface is very limited, making it hardly usable for scenarios where a larger amount of data items, for example the friends of a social network user, have to be displayed. Furthermore, the designs so far only give an overview of the privacy settings for exactly one recipient, for example the app manufacturer. However, in some cases like a retail scenario, there are multiple stakeholders (retailers, marketing agencies, suplliers, user's friends, etc.) that might be interested in the data and that each need their own radar interface, making it hard for the user to get an overview at one glance.

Finally, in-situ feedback is used in many domains, for example for gathering user feedback on newly developed or well-established software (like popular operating systems). So far, it has not been examined whether privacy violations can also be captured using in-situ feedback applications, for example on mobile devices. In particular, user's expectations on the effects on the privacy settings are currently unclear. In the next section, we will formulate the research questions arising out of the discussed problems.

1.7 Research questions

The central research question we try to solve throughout the thesis is:

How can we allow the user to better fit her privacy settings to her *individual* needs, without introducing additional user burden?

This research question can be divided into several subquestions:

RQ1 Can *personality* and *privacy* measures be automatically derived without additional user burden?

RQ2 How can we motivate users to carry out their friend grouping tasks?

RQ3 Can *personality* and *privacy* measures be used as an input to derive *individualized* privacy settings, when no other information about the user is available?

RQ4 Can the privacy settings of other domains be used for recommending privacy settings? Which domains should be used, and what level of detail should the settings have?

RQ5 Can we enhance radar interfaces to support large amounts of data items and multiple groups of recipients?

RQ6 Can we use *in-situ feedback* to capture a user's privacy violations as they arise, and what consequences do users expect?

RQ1 will continue existing work on deriving personality measures out of written text, which has been proven to work for several social networks like Facebook and Twitter, and also for comments on the video platform YouTube [104]. Nevertheless, there is no research so far investigating whether it is possible to predict *privacy measures* using written text. We will investigate whether there are correlations between the frequency of words belonging to a certain topic (like words about hobbies, sports or religious topics), which categories these are, and how well privacy measures in particular can be predicted with them.

To solve **RQ2**, we will investigate ways of enhancing the user experience when grouping data items, for example sorting friends into friend lists. We will especially focus on how the usage of new interaction systems and the associated new interaction possibilities can be utilized to increase the pragmatic quality, but most importantly the hedonic quality of a sorting interface. We will examine how we can enhance conventional card sorting metaphors with gamification elements in order to achieve the goal. By solving this research question, we will gather insights on which UI designs help to engage the user in the sorting task, and especially how the design affects the error rate as well as the needed interaction time.

With **RQ3**, we investigate whether, instead of the already discussed clustering approach, it is also possible to use personality as well as privacy measures (*individual measures*) to compute privacy settings which are unique to the user, based exactly on the given user's individual measures, instead of using one-size-fits-all privacy settings. We will investigate especially whether individual measures are factors that have a significant influence on the privacy settings, and whether those should be integrated into recommender systems that are so far only based on context factors. The approach presented should not be limited to one special domain like social networks, but should also be tested for use in other domains. We will thereby also investigate whether the impact and importance of the different individual measures

is always similar throughout different domains, or whether the most influential individual measures depend on the domain.

With **RQ4**, we are the first to study whether there are correlations between the privacy settings of different domains, and how they can be used for a cross-domain prediction of privacy settings. We will focus especially on the question of what level of detail is needed for a meaningful prediction, how precisely the approach can predict the privacy settings, and how much the precision can be increased when using more detailed privacy settings (for example using a set of privacy settings based on context factors like occasion, group of recipients, etc., vs. *one* general privacy level for a domain).

RQ5 can again be divided into two subquestions, one seeking improvements to support large amounts of data items, and one asking how multiple groups of recipients can be supported. For both questions, we will design user interfaces and evaluate their effectiveness in a user study. The special focus in the former question is how we can allow all data items to be displayed, while still allowing the user not to overlook any data item that is potentially important. That means we will evaluate whether our advanced design for displaying large amounts of data items allows a selection of data items with a reduced amount of false positives and negatives compared to the current traditional list-based solution, meaning it is less likely that a user will unintentionally select a data item that should not be disclosed (false positive) or fail to select a data item that should be disclosed (false negative). For the latter question, we have the goal to design a UI that allows a good overview on the privacy status, and that allows detecting unusual and potentially misconfigured parts of the privacy settings at one glance. The studies help us to understand how radar interfaces, and user interfaces with a limited space in general, can be enhanced to deal with the two aforementioned typical challenges of displaying and adapting privacy settings, and how effective the proposed techniques are.

Finally, **RQ6** will first investigate whether users see potential for in-situ feedback applications for capturing privacy violations at the moment they occur, regardless of the user's current location or occasion. The research conducted for RQ6 will especially focus on finding out what users expect as a consequence of their feedback, meaning which changes in the privacy settings they expect based on their feedback. There are many variables that are unknown at the moment, for example whether the feedback should only affect the privacy settings regarding the requestor that has received negative feedback, e.g. the requestor that should not have received the data, or whether similar requestors, for example the friends in the same friend group, should also be affected by the feedback. It is furthermore unknown whether the first feedback should instantly affect the privacy settings, or whether only the second, third or nth feedback within a certain timespan should have some effects. Also the actual effects on the privacy settings are unknown, for example whether the requestors access should be restricted only for the data item with negative feedback, or whether the requestor should be permanently blocked by the user. By solving this research question, we expect insights on whether the approach in general is suitable for capturing privacy violations, as well as guidelines on how privacy settings should be adapted, or in what other way the feedback can be used to assist the user in enhancing her privacy settings. The next section will outline the envisioned approaches used throughout this thesis to solve the aforementioned research questions.

1.8 Approaches and contributions

The proposed privacy framework will support the user in several touchpoints in what we call the *privacy journey*: Already today, doing privacy settings requires more than just defining the recipients when sharing a sensitive data item. A user also has to create groups of recipients in advance that can be used when sharing a data item later, or to adapt the privacy settings of a data item at a later point in time, as privacy desires may change over time. All these actions in connection with choosing privacy settings, including the adaptation of privacy settings, but also preparatory actions like the grouping of recipients, are what we call *touchpoints*, which are summarized in the *privacy journey* of a user. Figure 1.1 gives an overview on the touchpoints in the privacy journey that we want to address.

At the beginning of the journey, we have to create the user model by deriving the individual measures of the user, e.g. the personality and privacy measures (**RQ1**). Based on previous work on deriving personality settings from written text in social media, we will perform a user study to examine *whether* there are correlation between the word categories and the privacy measures, *which* categories have a significant correlation and are thus suitable for a prediction, and *how precise* a prediction of the privacy and personality measures can be using this technique. Through this research, we will get meaningful insights on how a privacy user model can be built without introducing additional user burden, and what precision can be expected from such a user model. Furthermore, we will gather information about *which word categories are good predictors* for the privacy measures, allowing us to speculate about correlations between a person's interests, writing style and privacy desires.

In parallel to creating the user model, the second touchpoint that we are addressing within this thesis and that has to be dealt with prior to giving privacy recommendations, is the grouping of recipients (**RQ2**). So far, research has concentrated on increasing the pragmatic quality of user interfaces and the automatic derivation of recipient groups, but has not yet found a solution. Within this thesis, we will examine how we can make the task of grouping recipients *more interesting* and thereby *motivating* for the users. Through this thesis, we will provide designs and insights on how new interaction principles like virtual reality and gamification can lead to an improved user experience within unchallenging tasks that are perceived as a burden by users. We will especially show and discuss how the gamification of this task influences the error rate and the time needed to do the task.

Having the user model and the recipient groups at hand, we will also support the user in the actual privacy setting task using recommender systems (**RQ3**). In contrast to earlier systems, we will focus on the individual measures as an input for a regression-based prediction of privacy settings. We will test our concept in four domains, namely the privacy settings for social network posts, shared locations, the permission settings for smartphone apps and the privacy settings for data from an intelligent retail store, like Amazon Go[15], that is equipped with sensors to record customers' movements and actions within the store. The research will produce new insights into whether personality, privacy measures, or both have a substantial influence on the privacy settings in the aforementioned domains, and whether their influence is similar across domains or dependent on the domain. Furthermore, we will acquire results about *which* individual measures have an influence on the privacy settings, and how precise a recommendation is possible using only individual measures. The work presented in this thesis can be combined with traditional

[15] https://www.amazon.com/b?node=16008589011 (last accessed: 2020-03-09)

context-based approaches as an additional domain of input variables to further increase their recommendation precision.

For the cross-domain privacy recommendation approach (**RQ4**), we will first conduct an exploratory study that tries to find indications as to which domains should be used, and which of the privacy settings are good candidates for a prediction. In a subsequent validation study, we will validate our findings by evaluating the precision of the recommender system based on the input measures determined in the exploratory study. The research within this approach will give insights on *which* domains should be used for the prediction, *which level of detail* leads to the best results and how far the precision can be increased using more detailed input (a set of privacy settings based on context vs. *one* general privacy level), and what *prediction precision* can be expected by the recommender.

Besides the recommender systems, a succesful privacy framework always needs a UI component in our opinion, allowing the user to manually review and adapt the recommended settings (**RQ5**). As stated above, radar interfaces have been proven to be a beneficial design to give the users an overview on their privacy settings, and furthermore motivate users in actively engaging with their privacy settings. However, they are unsuitable for scenarios involving a large amount of data items or multiple layers of privacy settings, for example for multiple groups of recipients. This thesis will advance radar interfaces in two ways: First, we will contribute techniques that allow radar interfaces to display a larger amount of data items within their limited space, making them suitable to display large amounts of data items, like the potential recipients of a social network post. Second, we will advance the radar design to support the display of multiple layers (for example for different groups of recipients) of privacy settings at a glance, allowing users to get a better overview and detect potential misconfigurations in all privacy policies at once.

Finally, we introduce a new touchpoint that has not been examined in research so far. In-situ feedback has been used for many other domains like software engineering, but the usage of in-situ feedback for capturing privacy violations as they occur has not been part of research so far (**RQ6**). Our studies deliver insights on whether the idea of in-situ feedback is perceived as beneficial by the users, what consequences users expect from the in-situ feedback, especially regarding their privacy settings, and how in-situ feedback can be further used for assisting users in managing their privacy.

Within this thesis, we solely address the threat model of unintenionally sharing data with a larger audience than intended by the user. We do not address other privacy threats such as unauthorized usage and propagation of private data by a platform provider, fraudulent acquisition of private data through hacking attacks, or other privacy threats existing in the online domain as discussed in Chapter 2. The influence of context factors has already been examined well in research in the past; therefore, we concentrate on the additional precision that can be achieved by including individual factors in the recommendation. Within the approaches mentioned above, we concentrate on finding out whether the approach meets the respective goals in the context of a lab study using real data sets and environments as much as possible. However, the thesis *does not* contain in-the-wild studies, where the solution is integrated into an operating environment like a well-established social network, and elaborated through the daily usage of a large number of users. Thus the results regarding the precision of the recommender systems and the user acceptance of the presented approaches can be seen as an approximation towards what the approach could achieve within a running environment.

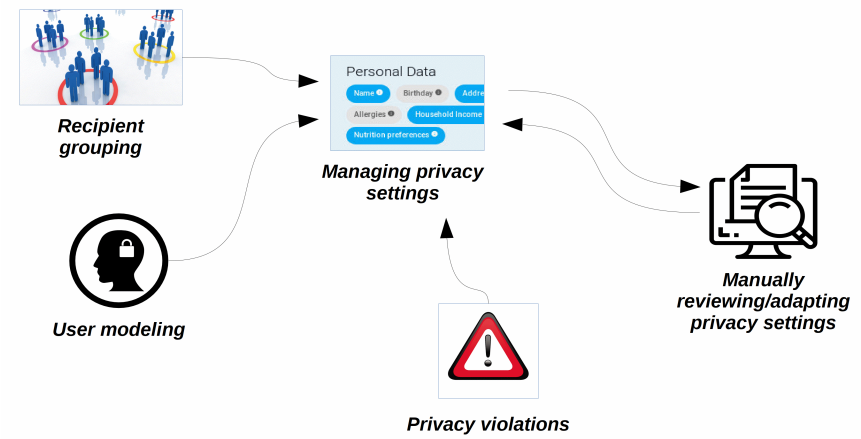


FIGURE 1.1: Touchpoints in the privacy journey addressed by the proposed privacy framework.

1.9 Thesis outline

In the next chapter, we will first define several terms important for this thesis, and give the background knowledge needed for the remainder of this document. We will especially show problems that users have with the current privacy controls, the privacy threats that arise out of these problems, and the threat model we are addressing. We will discuss related work on the context factors and individual factors that have been discovered by research so far, and existing user interfaces and privacy management systems and their drawbacks. Finally, we will give a short introduction into machine learning techniques and cross-domain user modeling approaches, which constitute the basis for the approaches of the recommender systems presented in this thesis. Before discussing the recommender systems, we will first show how a user's written text can be used for deriving a user's individual measures (**RQ1**)based on text analysis and machine learning in Chapter 3. Chapter 4 will then try to solve **RQ3** within the social web: to be more precise, for recommending the audience for a social network post and the precision of a location to be shared with the friends of the user. The subsequent chapter then validates that the approach can also be applied in other domains outside of the social web, using the example of recommending permission settings for smartphone apps and privacy settings for the private data captured inside an intelligent retail store. Chapter 7 examines the derivation of privacy settings using the privacy settings from other domains (**RQ4**), concluding the treatment of recommender systems proposed in this thesis. Chapter 8 discusses the user interfaces that are part of the privacy framework, and thereby addresses several research questions. The first section addresses how users can be motivated in carrying out the task of recipient grouping through enhancing the user experience of the task (**RQ5**). Afterwards, we will introduce OmniWedges and URetail, two advanced radar interfaces that allow the display of large amounts of data items and multiple sets of privacy settings for multiple recipients (**RQ2**). The chapter ends with a study regarding the usefulness of in-situ feedback on privacy violations as well as possible applications for the gathered feedback (**RQ6**). Finally, the thesis closes with an outlook and conclusion, summarizing the thesis, the key contributions, future work and limitations.

Chapter 2

Background and Related Work

Privacy and privacy settings are not something that can be handled once and then remain the same for the rest of a person's life. They are a process which is steadily changing over time, over the years and the life experience of a person [233]. Therefore, privacy user interfaces should also be tailored so that the privacy setting process is not done only once, using a wizard when the user enters the system. Instead, privacy settings have to be checked, reviewed and adapted continuously [233]. Sometimes, it is also necessary to actively engage the user in reviewing the privacy settings from time to time to achieve good data privacy [333]. As a logical consequence, privacy user interfaces should, on one hand, avoid methods requiring excessive configuration and maintenance, but on the other hand, they should still give users fine-grained options to control each aspect of their privacy [200], which makes the task of choosing privacy settings very hard not only for users, but also for researchers. At the beginning of this chapter, we will discuss current privacy problems of lay users in different domains, as well as the potential risks associated with these. We will then review how privacy (settings) management is currently handled in research, and which input data, e.g. context factors and individual factors, have been found to have an influence on privacy decisions, and which could thus be useful to perform a personalized prediction of privacy settings. After discussing the different approaches in privacy user interfaces that support the user in one way or another in increasing his privacy, we finally review current techniques in machine learning that will be applied to make privacy setting predictions, as well as related work in the domain of model transfer, which will be the basis for the development of privacy setting prediction using privacy settings from another domain (Chapter 7).

2.1 The lay user and her privacy problems in social media and other domains

Since the world wide web emerged, the transfer and retrieval of information has become significantly faster and easier [231]. On one hand, this eases the transfer of data and knowledge; on the other hand, it can also be a threat to people's privacy. A special problem in this situation is that it is nearly impossible for the data provider to foresee for which applications and re-applications his or her data will be used by others [231]. Therefore concerns about privacy are totally legitimate in a computerized society, and are a key performance indicator for privacy. A society in which privacy and therefore security is not present cannot develop and will not be sustainable [231]. It is therefore important to create zones of privacy for each person, for example in the home environment, so they can plan their lives without fear. In those environments, people can restrict their privacy toward different individuals

according to their needs. This holds for their physical home environment, as well as their "digital home", e.g. their social media pages and private data [231]. Most people share their data with their own perception on how far the information should spread, which is mostly based on a social norm that they expect of the information recipients. However, this social norm is often violated by the information recipients, which results in a privacy breach as perceived by the original poster [237].

2.1.1 Audience misperception and misunderstood privacy regulations

Often, such a privacy breach is caused by a misperception of the actual audience of some piece of private information, which is often multiple times larger than the perceived audience. On the social network Facebook, for example, studies have shown that the perceived audience consists of only 27% of the actual audience, which means the actual audience is four times bigger than users think when publishing their posts [34]. This is especially problematic as only a small subset of Facebook friends are real friends; it happens very often that friendship requests are accepted so that the requesting person will not be upset, although the user does not see him as a real friend with whom she would like to share her private posts [66]. Users often accept even friendship requests from users whom they do not know at all: In a study by Gross and Acquisti [139], 250,000 users were sent a friendship request from a user made up by the researchers, who was unknown to the participants. Out of those 250,000 users, 75,000 accepted the friendship request. Later studies in 2014 confirmed that this still holds, if the friendship requestor has the name or profile picture of a known friend, lives in the same area, or shares common interests with the user [271].

There is also a large deficit in understanding of the terms and conditions and privacy policies of such social network providers: A study involving Facebook users [327] found that 85% of the participants did not read any part of the terms and conditions of the service provider, and were therefore not aware of what data is collected by the provider, how it is processed, and for which purposes, including online marketing, it can be used. 79% did not read the privacy policy that gives more detailed information about those privacy aspects of the terms and conditions. This results in the fact that 73% of the participants were not aware that Facebook is using their private data for marketing purposes. More than half (55%) did not think that Facebook apps send data to the companies that developed them; they thought the apps were running isolated on the Facebook page and had no connection to a server belonging to the developing company. Finally, 63% of the participants thought that the shared information was only visible for the accepted Facebook friends although this is not the standard setting; data is shared to friends of friends as well per default.

2.1.2 Narrowcasting – solution or additional challenge?

Often, social network users do not use privacy settings at all and leave them at the standard settings [355, 174]. Instead, they use a *restriction strategy*, meaning they only publish information which they do not see as critical [355], although more sensitive posts might also be of interest for close friends and can increase social ties to those friends [98]. Another strategy similar to the restriction strategy is the *narrowcasting strategy*. Instead of restricting the content that is published online, the narrowcasting strategy consists of restricting the recipients of the information [134]. Apart from increasing a user's privacy while keeping the utility of the service and the positive effect on social ties at the same level, narrowcasting also reduces the message load

for the user's followers and friends, so that the information they receive in their feed or on their news wall is reduced to the topics that are of interest for these specific persons. Currently, Facebook is using a kind of algorithm called *algorithmic curation* to automatically organize, select and present the posts of a user's friends that should be displayed on the news feed. However, if that is done in the background without notifying the user (as it is done on Facebook), most of the users (62.5%) are not aware that such a selection and ranking is taking place. When users find out about this, it often leads to anger and frustration, as they feel patronized by the social network providers, or have the feeling of being manipulated by them [270, 100]. In contrast, if a website includes a user interface where the user can see and adapt the curation algorithm to his needs, this leads to significantly more active engagement with the social network privacy settings, and increases the feeling of control and privacy on the site [100].

Social network providers have tried to increase the privacy control of their users by introducing *friend groups* (Facebook) or *circles* (Google+) that have to be created by the user, and which can then be used for a narrowcasting strategy, so that only friends and social circles which are trusted by the user and which are interested in the information will receive the post. However, creating the friend lists introduces a major user burden [247] and therefore, users rarely use them [276]. Social networks also offer automatically generated friend lists; however, these friend lists rarely reflect the friend groups that users need and that they would create manually [230]. Study results on the prediction are often biased, as they only ask users whether the created lists are acceptable, rather than checking whether the created lists reflect the actual friend groups a user would create from scratch [230]. If compared to user-created lists, none of the machine learning approaches, based on the friends' profile data, friendship graph or activity of the friends, leads to an acceptable solution [230]. That might be the reason why only 67% of all Facebook users use friend lists, and only 17% of all Facebook posts are shared using friend lists [230]. More than half of Facebook users have not created any friend list, and about 15% and 10% have created only one or two friend lists, respectively, which does not allow for a meaningful narrowcasting strategy [167]. Especially pictures, which often contain upsetting details, are shared without the usage of friend lists in more than 80% of all cases [167].

At the beginning of social networks, those networks were concentrated on small social circles, like a university community (Facebook); therefore, the privacy threat was smaller then than now [86]. In a longitudinal study from 2005 to 2011 involving several thousand Facebook users [316], researchers found that the amount of publicly shared data was continuously decreasing, while the actual amount of written posts was increasing exponentially. Facebook reacted to this by a change to the UI, where more of the privacy settings were set to public by default. Still, the privacy settings do not affect the data that is shared with Facebook and its advertising companies for marketing purposes, which includes *all* data entered on the social network platform, according to Facebook's terms and conditions. However, this fact is known only to a minority of users [316].

2.1.3 Google+ and its advanced privacy UIs

Some years after Facebook became the most popular social network worldwide, Google started to create a competitive product named *Google+*. Although the concept is very similar to that of other social networks, users behaved slightly differently, including with respect to their privacy decisions [219]. Google+ has been found to be more popular in countries with a lower internet penetration rate, thus

FIGURE 2.1: Google+ circles used to group social network friends. Image source: https://www.rivaliq.com/blog/making-google-marketing-work-for-you/

being used more by people in developing countries that are not as tech-savvy and privacy-aware as users in industrial countries [219]; therefore they shared more sensitive data. Until the shutdown of Google+ in 2019, the average path length between users was higher compared to Facebook, which was another possible reason why Google+ users felt better about sharing sensitive data, as information spreading is slower in this network [219]. Apart from the different user base, Google+ was the first social network that put the creation and management of friend lists, called *circles* in Google+, in the center of the user interface (see Figure 2.1). The user's circles were shown as a part of her own social network page, including several pre-defined circles and a UI that was thoroughly optimized towards usability. This design choice also had an impact on the usage of friend lists, which was more frequent in Google+ compared to its competitor Facebook [177]. In the first days of Google+, about 75% already used selective sharing at least once a week, whereby 67% of all items were shared using circles.

2.1.4 Privacy problems in other domains

Smartphone app permissions

Similar problems can be found in other domains as well. Concerning smartphone app permissions, studies have found that only 17% of all users pay attention to the list of permissions that are required by an app upon installation; the remaining majority chooses to install the app immediately without a review [109]. This effect might also be caused by the technical knowledge that is required in order to understand the permissions that are required, which data they allow an app to generate, and which consequences might occur for a user if she enables that permission. Only 3% of users claimed to understand the information shown on the permission screen, indicating the lack of technical knowledge and lack of information on the permission dialogue as one of the possible causes for the low privacy awareness of users in this domain [109]. Researchers have proposed several solutions to this threat, like

distributing the tasks that need access to private data over several decoupled online service providers (anonymous task distribution), privacy-aware data processing, data retention, and the possibility for the user to control and audit the access to her private smartphone data, among other approaches [68]. However, these countermeasures always need to be implemented by the app manufactures, meaning that end-users cannot address the problem by themselves.

Location sharing

A similar situation can be found in the location sharing domain. If privacy controls are present, they often have poor usability that confuses end-users, making it hard for them to understand their benefits. Users therefore often do not touch the location privacy settings at all [325]. However, research has found that users actually *can* understand the mechanics and effects of location obfuscation techniques, if they are explained well [51]. Furthermore, users can also detect if an app requests a permission that is obviously not needed by the app, for example if a flashlight app is requesting access to the contact list [325]. Interestingly, users are also willing to trade their privacy for cash. In a representative study involving 32 smartphone users, researchers found that users will trade their location data for one month for as little as $100 on average [51].

Online shopping

Finally, this privacy problem is also present in an online shopping scenario, especially when it comes to product recommendations that are initiated by the customer. A study found that in this case, customers willingly answer almost all questions, even if there is no direct connection between the desired product type and the questions asked, making it possible to extract private data from the customer that can later be used for various attacks like social engineering (see next section for details) [310]. It has further been shown that the usage of eBay by itselfcan lead to a privacy breach. Research has shown that the rating portal of eBay offers enough private data to link an eBay account with a corresponding Facebook profile, so that the customer's personality profile, his private interests, workplaces and his circle of friends can be connected with his items bought on eBay [229]. The researchers were able to correctly connect 17% of the eBay profiles of the users taking part in the study.

To conclude, we can say that severe privacy problems exist in several domains, and that these have to be solved in order to prevent further negative consequences. In the next section, we will discuss which negative consequences misconfigured or neglected privacy settings can have in the digital world. As we will see, it is not only *oversharing*, i.e. sharing too much information that can be seen as a threat (to privacy). *Undersharing*, i.e. sharing less data than possible for maintaining a user's privacy, also has negative effects, as it is a threat to the user's socializing and the social capital of a society in general.

2.2 Privacy risks and benefits of the social web and threat model

In the media, the social web, and especially social media websites like Facebook and Google+, are usually known for bad effects, like unintentionally sharing sensitive data, hacking attacks, privacy violations and the loss of social contacts in the real

world. Although studies have shown that those negative aspects do exist, there is also a high amount of positive effects arising from the usage of the social web, as the following two subsections will show.

2.2.1 Privacy risks in the social web

Most users do not share their whole social network profile with the public; only a small amount of the information can be seen by everyone, whereas most information, especially sensitive data items, is shared with *all* friends regardless of whether it is a close friend or just a friend on the social network that the user has never met personally. However, even items that do not look critical at a first glance can be used for various attacks [283]. For example, email addresses are crawled and stored by attackers, either to sell them for spam campaigns, or to store them in a database, so the data can be used to attack that specific person later[1]. Therefore, hiding or removing sensitive data items afterward is not necessarily a meaningful countermeasure to future hacking attacks, as hackers store such data on separate servers once it has been retrieved. Moreover, as the data retrieved from a social network usually contains other private information, like birthdays, places lived, and interests, the data items are significantly more valuable than usual, and will retrieve a higher price on the black market compared to a data set containing only email addresses. Also, a date of birth seems unproblematic at a first glance, but can be used to confirm a person's identity over the phone, for example to change contract details or to make a new contract with a third party[1].

Hacked accounts, friend attacks and phishing

There have been several security breaches where Facebook accounts have been hacked, even those of prominent persons like Facebook founder Mark Zuckerberg [118]. There exist several tools that make it possible to take control over a social network account [225], for example by retrieving Facebook URLs of Facebook users that are logged in on the same unsecured network, using a packet sniffer, also known as *session hijacking* [160]. Once the account is hacked, it can be used for many attacks, especially on the direct friends of the victim. Social networks and their linkability provide an optimal breeding ground for spreading malicious links to viruses among the friends of the victim [225], who then spread the malicious software to their friends, and so on. Or instead of spreading malicious software, the data retrieved from a social network account can be used for making phishing attacks more efficient, for example by sending messages to friends, which include personal information that should be known only to the victim or her close friends, asking for the friend's private information, redirecting to a cloned site of the friend's online banking account, or asking her to install a malicious piece of software or Facebook game app so that they can "play the game together" [143].

Privacy threats caused by Facebook apps

Facebook apps are especially critical, as they are, contrary to what users think, not a part of the Facebook website, and neither hosted nor controlled by Facebook. These apps are websites of third-party developers that are hosted on the developers' hardware, and use the Facebook API to connect to the user's account and download the

[1] https://www.nfpcar.org/FOIA/security_guide_to_social_networks.pdf (last accessed: 2020-03-09)

requested information on their own servers. Facebook provides only minimal security controls, so that while the app manufacturer should actually use the requested data only inside the app, technically he can also store and process it without any limitations [291]. A study has found that more than 90%of Facebook apps actually do not require access to any private information; however, most of them still request access to it, in order to process the data and place targeted advertisements inside the app or elsewhere on the web [110]. Apps often also request permission to publish posts or write messages on the user's behalf; officially this is to publish achievements on the profile page, or to invite friends. Unofficially, this functionality is also used by malicious apps to write messages to friends that look like they were genuinely written by the user, but that in fact are generated automatically by the app in order to convince the user to click on a malicious link or provide sensitive information [110].

Identity theft and social engineering

Even without hacking a social network account, the privacy threats are numerous. As stated in the last section, when unknown users are sent friendship requests, about 30% of them accept the request even though they do not know the user [139]. As more recent research from 2014 has shown, this is still the case if the profile name or picture matches the name or picture of a real friend, if the requestor lives in a nearby area or if they share common interests, for example [271]. It is therefore rather easy to become a friend of the target person, or a friend of a friend (FoF), and to access the sensitive profile items which are shared to friends and FoFs per default. Using publicly available information, or also using the private information if the attacker was successful in becoming a friend or FoF, the attacker can create a fake identity based on the gathered information, which is also known as *identity theft*. Using this fake identity, the attacker can carry out criminal acts in the name of the victim and make contracts in her name. Often, it is also easy to answer the security questions of an account with the gathered personal information, and thereby to take over the victim's existing accounts. Identity theft can also be used in social networks to attack the friends of the victim, for example using two different techniques discussed by Bilge and Strufe [36]. In the first approach, the attacker clones the victim's profile on the social network, and sends friend requests to the victim's friends. If the friends accept the request, the attacker can again use the account to spread malware and perform phishing attacks, as described above. In the second approach, the victim is a member of a social network, but does not have an account on another social network. At the same time, some of the victim's friends have an account on *both* social networks. The attacker then extracts the victim's information on the first network, and creates a fake account in the second social network, where the attacker again tries to befriend the friends that are active in both SNs, and performs the aforementioned attacks upon success. In a study, 60% of a user's friends accepted the friend request of the cloned identity in the same social network, and 50% of the friends accepted it if the identity was duplicated to another social network.

Inference attacks

Apart from identity theft using the available data, it is also possible to infer some of the missing information data ("inference attack"). A study comparing different inference methods has shown that it is possible to reconstruct most of the personally identifiable information in most cases. The person's birthday could be inferred from

the account in 77-93% of all cases (depending on the algorithm), their hometown in 32-40%, their political views for about 50% of the profiles, and the partner of the person in 20% of all accounts that took part in the study [6]. Especially the birthday, hometown and current residence are of high interest for attackers, as they allow them to guess the social security number, which is used for identification for several legal acts [6]. It is also possible to reconstruct the friend list of a user, even if she has restricted access to it, allowing attackers to infer the social circle, social status and the possible income of a user [171].

Stalking

Stalking is also a typical problem which occurs very frequently on online social networks, especially as it is rather easy to become a member of the FoFs and have access to the profile of the target user, especially the posts and check-ins, so that the attacker always knows where the user is spending her free time and what event she is attending [139]. Also, re-identification of a person is possible, either by comparing an image of the person with the profile images of a social network, or by comparing and matching demographic data that is available for most users [139]. Unfortunately, suppressing profile items, even so that only friends, or nobody, can see them, helps only partially to prevent the aforementioned attacks: An inference algorithm developed by Altop and Nergiz showed some years ago that even in this case, it is possible to infer the personal profile items with an astonishingly high precision [11].

Sybil attacks

Similar to identity theft, the *sibyl attack* is also based on accounts or identities that have been taken over by the attacker [12]. A sibyl attack requires first gaining control over a *large amount* of identities in the network. The attacker can then connect those identities together, so that the attacker has control over a substantial fraction of the system, for example for manipulating the outcome of an election in the social network [12]. There exist countermeasures for a sibyl attack [299]. However, if the attacker manages to take over or insert fake identities so that at least one third of the system is controlled by the attacker, even those countermeasures will fail [299]. A sibyl attack can also be used for large-scale spam attacks, where each of the controlled identities is used to send spam to the friends of that node, and eventually also to other nodes that are not directly connected to the controlled accounts [160]. Furthermore, the controlled accounts can be used for a *puppetnet attack*, where the controlled accounts send messages and posts to other users which include links to images or resources of a specific website, so that displaying and reading those messages creates a massive workload on the target website servers, which then eventually crash or become unavailable, similar to a Denial of Service (DoS) attack [118].

User deanonymization and friendship network inference

Apart from the personal data of a user, it is also possible to infer the network structure of a social network, or to identify anonymous users [29, 186]. Studies have for example shown that if a user uses his real name and personal information in one social network and a fake identity in another network, it is possible to match those accounts through the structure of the social graph in both networks [29]. This is also possible if only fake or anonymized accounts are present, without information from another social network with real accounts, by taking the structure of the friend

network and publicly available background information into account. This is also known as an *attributed couplet attack* [186]. Hackers as well as social network sites also use *link mining* techniques which analyze the friend graph in order to find missing links (to propose new friends to the user), clustering users into groups, as well as collective classification of the clusters found, allowing them to assign the users to different user types [127].

Location inference

Privacy problems can also be found in other domains like location sharing [193, 354, 279]. Users are often unconcerned when sharing their location, although it has been shown that always sharing one's location even without any further information makes it possible to infer the home address of the user [193]. It is also possible to infer a user's residential address simply by analyzing the *likes* on a social network page [354]. In the algorithm of Yamaguchi, Amagasa and Kitagawa, the likes are used to identify landmarks that are shared by users (for example local radio stations) which can then, based on their location, be used to narrow down the home addresses of the connected users [354]. Conversely, it is also possible to infer the user based on the check-ins she typically does on the social network page. In a representative study, researchers were able to train a machine learning algorithm with a data set containing both user names and check-ins from the past to correctly identify the users based on their check-in behavior in the future in about 30 to 50% of all cases on Foursquare and about 80% on Gowalla [279].

Another vector that can be used for attacks is comments and posts that users publish on their profile page, or more precisely, the writing style of the user [8]. Using text analysis and machine learning, researchers were able to correctly match 90% of all participating accounts between two social network platforms. However, there exist very effective countermeasures to this attack, like rewriting the comments in a crowdsourcing approach, or simply translating them to different languages and then back to the original language [8].

2.2.2 Benefits of the social web

However, although there exist many risks when taking part in a social network community, actively using social networks also has several positive effects for the user and society [98, 146, 337, 26]. Most users use social networks for a specific reason and not only as a pastime [3]. Users are interested in creating a positive picture of themselves, and to become more accepted and admired in their social circle [3]. The general opinion is that spending time online reduces the amount of social interaction, especially face-to-face interaction, and weakens social ties, leading to a diminished social capital for the individual [236]. However, studies have shown that this is in general not the case [25], as it depends on the actions done online. The problem is that early research concentrated on the loss of social capital in offline communities due to the increased time spent online, but did not take into account the gain in online social ties that might compensate for this effect [341].

In fact, it can even help to tie new communities together and increase one's social capital [25]. Social networks can help retaining social bonds, especially to friends from the past (like school friends) or friends living abroad, making them the perfect tool for maintaining long-distance relationships [98]. Social networks greatly strengthen such weak ties in particular, as maintaining them over a long distance

is cheap and easy compared to face-to-face meetings or telephone calls [92]. Therefore, researchers claim that they can supplement or replace in-person interactions, mitigating any loss from time spent online [338]. The internet supports offline networks through online communities, where the information technology can enhance the place-based community, as it is possible to connect and discuss with other individuals on any day, at any time, leading to an increased social capital [146]. Communities are no longer restricted based on their geography, but can form based on shared interests, regardless of the hometown or nationality of the person [337]. It has also been shown that a large amount of online users that met online are meeting their correspondents face-to-face later [244]. Parks and Floyd stated in their work that "relationships that begin online rarely stay there" [244]. Social media, especially locality-based social networks, can increase the interaction of the local community and motivate people in taking part in discussions [146]. Members of online communities are also more likely to receive help from others, and in fact have a larger network of close ties than non-internet users [41].

Furthermore, studies have shown that especially persons with a low psychological well-being can profit from an online community, as it is hard for them to form offline relationships with friends and neighbors, while it is easier to be part of an online community, for example by starting to play games together and finding friends within this context [25]. Online communities also have lower communication barriers and ease interaction with others, thus encouraging self-disclosure, and allowing especially younger individuals to construct up their personality and their self-view [26].

As we can see, there are many potential risks when using an online social network. Of all the mentioned risks, our work tries to reduce the amount of *unwanted disclosures* made by the user and her chosen privacy settings, e.g. the privacy threats that emerge through *oversharing*. Oversharing can ease social engineering as well as inference attacks and stalking. However, reducing the amount of shared amount to zero, e.g. not taking part in any online community, would lead to the (social) threat of *undersharing*, leading to a loss of social ties and social capital for a society. *Therefore, the goal of our work is to reduce oversharing (caused by misconfigured privacy settings) and the privacy risks connected to it to a minimum, while keeping the positive social effects of online communities intact.*

However, as we have seen reducing the amount of shared data does not lead to *perfect security* from all attack models. There are also attack models that allow inference attacks and social engineering when some data items are restricted by the user. But to be honest, perfect security does not exist. Even if countermeasures for those attacks are included and a system is created that includes the state of the art to avoid all of the privacy and security risks known so far, such a system would still not be safe. There would be still privacy and security leaks that have not been discovered so far, or for which effective countermeasures do not yet exist.

2.3 Privacy management systems

Studies have shown that sharing private information with a limited audience not only increases the privacy, but also makes the content that is shown to social network users more interesting, as the authors manually choose recipients that may be interested in the written post [134]. Even simple approaches involving age, location and gender as selection mechanisms lead to an improved privacy and usability of

a social network [134]. Furthermore, using narrowcasting has been shown to increase privacy awareness, while the same amount of content is shared, but only for a targeted audience [134].

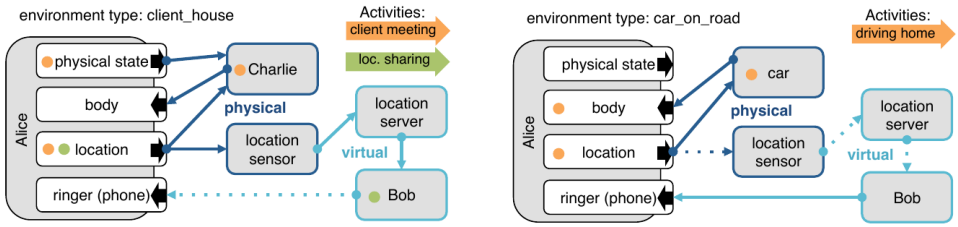
Typical systems supporting the user in doing narrowcasting, e.g. in automatically setting the privacy settings for the new post correctly, often rely on user settings chosen in the past [296, 295, 31]. However, it has been shown that users' privacy desires and privacy attitudes, i.e. the privacy level a user *wants* to have, often differs from the privacy settings that he *actually* chooses; this is also known as the privacy paradox [27]. This effect has been observed in online communities and online social networks as well as other domains like e-commerce [310]. Research has tried to identify the causes for this effect. So far, results indicate that it is hard for users to translate their privacy desires into actual privacy settings, as they lack the technical background knowledge to do so [218]. The study also showed that users are usually not aware of the misconfiguration of their privacy settings [218]. It has also been shown that in addition to a lack of knowledge, users are not willing to invest the time to adapt their privacy settings correctly and often do not understand the implications and consequences of their privacy settings, especially the consequences that arise if they do not adapt the standard settings offered by the provider [128, 218]. Furthermore, it is hard for users to assign their friends to friend groups, which is a basic requirement for narrowcasting [336]. Finally, users often also have to deal with usability problems regarding the general workflow, UI mechanics or simply user interface-related problems [179, 346, 184]. To conclude, if a user's previous privacy settings are used as the only source for proposing new privacy settings, one always has to keep in mind that the settings from the past can only be seen as a rough approximation of the actual desired settings, rather than a gold standard.

2.3.1 In the social web

The privacy management systems existing in research for the social media domain can be divided into different types of systems: *rule-based* systems, *machine learning-based* systems and *encryption-based* systems.

Rule-based systems

Rule-based systems do not contain any kind of artificial intelligence, but rely on more or less complex rule sets that are created by the user, either directly using a rules file or indirectly using a user interface. A simple example of such a rule-based system is the work by Carminati, Ferrari and Perego [58], where access to a social network resource is granted based on the path between the original poster and requester in the social path, and the *trust levels* between the users on the path. The trust level is given by a user for all of her friends (i.e. all outgoing edges from her node in the social graph) in the interval $[0; 1]$. In addition to this, the user can specify whether he fully trusts the user, or only partially. Based on these factors and the length of the path from the original poster to the requester, the poster can create rules for which users can access the post, and which users cannot. A similar but more complex tool is the mobile access control list (MACL) [215]. In order to use this system, the user has to first fill out a questionnaire, which is used to generate a default setting. The user can then tune the settings based on his needs, by applying custom rules for particular contacts and also contexts (like "at work", for example). Based on this rule set, the MACL offers three different functionalities for location sharing. First is *long-standing location sharing*, where the location is always shared,



(a) User (Alice) is at client (Charlie).

(b) User (Alice) is driving home after work.

FIGURE 2.2: Rule-based privacy management system involving context factors [284].

but the precision of the shared location depends on the geographical distance between poster and requester. If the user is in the same city, the requester is shown the original location, while she can see only the name of the city if she is outside of the city, for example. The *rendezvous* functionality shares the location only with specific users at a specific time of the day, for example for meeting up to go out. Lastly, the *proximity detection* functionality shows up if one of the user's close friends is nearby, so they can meet up. From the three proposed functionalities, only the rendezvous functionality was rated as useful by the majority of the users. Another approach adapting the idea of context-based privacy settings uses the activity of a user for defining the privacy settings with the example of a traveling salesman [284] (see Figure 2.2). If a saleswoman (Alice) is on the way to a client (Charlie), the saleswoman's supervisor (Bob) may see her location, and also send her messages or make phone calls. When she has arrived at the client, all incoming notifications are muted, and the location may still be shared with the saleswoman's supervisor. After leaving the client and driving back home, the saleswoman is in her leisure time; therefore, the supervisor is neither allowed to write messages nor send phone calls, nor can he see the saleswoman's position. The aforementioned systems all let the user directly create the rules, which requires some technical knowledge. In contrast to that, other systems allow users to specify the privacy rules in free-text form using their natural language [49]. Using text analysis tools, these rules can then be transformed into an xml-based file, which can then serve as a basis for the aforementioned rule-based tools.

Machine learning-based systems

Whereas rule-based systems can be distinguished especially by the complexity of their rules, machine learning-based systems differ significantly in the algorithm on which the system is based, and therefore also the input that is used for the prediction of the privacy rules. One approach of machine learning-based systems is to use a subset of friends labeled by the user regarding their trustworthiness, which are then used to derive labels and the corresponding privacy settings for the remaining users [296, 295]. The same can also be done interactively by observing the "allow" and "deny" decisions of a user as he publishes new posts, to iteratively train the machine learning system and to give recommendations when the algorithm has reached a certain level of certainty within the predictions [31]. Whereas the aforementioned papers did not distinguish between the post content for predicting a privacy policy, Sinha, Li and Bauer [300] used NLP techniques like Latent Dirichlet Allocation (LDA) and Maximum Entropy to predict the privacy setting of the post for the different friend groups. The system is based on a supervised approach, which uses

the sharing settings of former user posts in order to train the predictor. Although this method spares the user from giving explicit feedback to train the system, incorrect privacy decisions made in the past will be used to train the system, resulting in prediction errors. Furthermore, we believe that in the mental model of a user, it is not the post content that correlates to a privacy setting, but the *topic* of the post. While the aforementioned approaches suffer from the privacy paradox as they rely on privacy decisions made in the past, other systems train their machine learning model using explicit user feedback, for example by displaying access requests to the user and using the feedback to train the algorithm, which can then, when the certainty of the algorithm is good enough, be used to reduce the amount of access requests that are displayed to the user [350]. Another possible input that has been used in research is the actual post content for which the privacy setting has to be generated. Using NLP techniques like Latent Dirichlet Allocation (LDA) and Maximum Entropy, it is also possible to offer privacy settings for the user's friend groups [300]. The proposed method is based on a supervised approach, using the sharing settings of earlier user posts to train the machine learning model. Lastly, some of the approaches also involve a questionnaire at the beginning, which helps the system to classify the user and assign her an initial privacy profile, which can then be adapted by the user [103]. After the system has automatically grouped the user's friends, the system uses the group memberships of each user as an input to predict privacy settings for each of the friends using a machine learning classifier.

Encryption-based systems

Lastly, encryption-based systems concentrate more on data privacy against the social network provider, rather than data privacy against other social network members like the aforementioned approaches. These approaches are mostly based on encrypted data transfer between end users, so that only the desired recipients of the information can decrypt the data. "Persona" is a social network that is built around this idea of encrypted end-to-end communication between users [20]. In this social network, the user can create custom friend lists by hand, and share the data only with the selected friends and friend lists using an encryption method which allows only the selected recipients to decrypt the post. A different and more generic method is proposed by Beato and Kohlweiss, and is realized as a browser plugin [57]. This plugin (see Figure 2.3) allows users to publish posts on a social network site, which is automatically encrypted by the plugin. The recipients of the post can then only decrypt the message if they have installed the plugin, and if they are chosen as recipients of the message. Using this technique, nearly every social network website, and also parts of every other online website, can be encrypted independent of the actual website owner. On the other hand, only users that have the plugin installed are able to see the post, even if they have been chosen as recipients by the original poster.

Custom systems for special use-cases

There are also systems that target special privacy scenarios, for example how the privacy settings can be derived for online content which belongs to multiple owners (for example a photo with several persons in it) using an auctioning system and game theory [312]. Other approaches try to solve the privacy problem from the side of the data collectors, by raising their privacy awareness, so that they do not collect data accidentally when the data owner did not allow it. The user, on the other

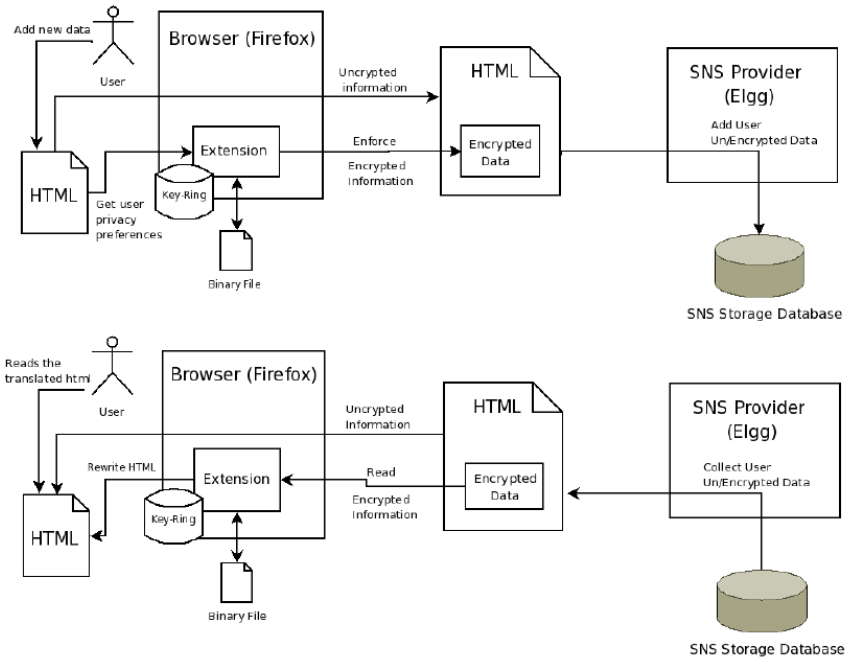


FIGURE 2.3: Architecture of an encryption browser plugin for securing sensitive social media data [57].

hand, can see which data was accessed, processed, and deleted, so he can give active feedback on a privacy violation to the data collectors [197]. However, in the scope of this thesis, we concentrate on deriving privacy settings using a machine learning approach, as it allows the highest degree of automation and does not necessarily need technical knowledge from the users, as they do not have to specify access rules themselves.

Whereas the approaches presented so far all relied on context factors like time and date, or activity of the user, there are also some approaches that rely on the user's personality in order to provide a more personalized privacy recommendation. Most of them use a *clustering and template* approach, where each user is assigned to a group of users with a similar personality, which are then assigned a privacy profile that is tailored to that group of users. One approach following this idea has been proposed by Ghazinour and Matwin [128] and assigns the user to the three personality groups according to Wiley. After the assignment to the group, the algorithm searches for the three nearest neighbors of the user's profile, and the user is shown a summary of current differences and possible misconfigurations compared to the reference profiles. The approach by Guo and Chen, on the other hand, takes a different view on the user's personality and privacy desires [141]. In their idea, the user's privacy measures are the fraction of profile attributes that are shared and the profile items that are not shared either with friends or the public. The system has been trained with a data set containing both privacy measures and privacy settings. Whenever the privacy settings for a new user have to be predicted, the setting uses the trained classifier to find similar user profiles, and proposes the privacy settings of those similar profiles to the user. The user is then shown the predicted results together with a privacy description for each of the privacy settings, so he can then review and adapt them.

2.3.2 In other domains

Mobile app permissions

Proposing privacy settings is also a recent topic in other domains, for example in the mobile phone domain, where the user has to define and adapt the permission settings for all of the apps installed on her smartphone. Similar to the work presented above, those systems can also rely either on explicit user feedback, on the privacy settings chosen in the past, or other techniques like crowdsourcing. A simple approach by Liu, Lin and Sadeh creates a user profile based on the some example privacy settings for some permissions of some of the installed apps chosen by the user. Using a support vector approach, the system then derives the remaining permission settings in an interactive way, allowing the user to adapt settings, and using the input again for further training and adjusting the prediction algorithm. The proposed approach achieved an accuracy of up to 87% in the evaluation. A system by Liu et al. uses active user feedback instead of privacy settings by giving the user an overview on their apps and how often they used the permissions they requested (Figure 2.4), asking the user whether he feels comfortable with this behavior or not [209]. Based on this feedback, the approach recommends the privacy settings for the apps. In a user study, the participants accepted 78% of the recommended settings. Nevertheless, the approach needs information about the usage frequency of the permissions for each app, which is currently not available on a typical smartphone without extended technical knowledge. Other approaches, in contrast, do not need access to this information, but use static code analysis to derive the purpose of each requested permission [206]. Based on the user's reported comfort with the purpose of this permission, the system clusters users into groups with similar privacy desires, and generates a finite set of privacy settings, one for each of those clusters. Interestingly, if enough sample data is present, the prediction needs only a little context information to achieve a good precision. In an approach using a large online database of the LBE Privacy Guard app, containing permission settings of more than 4.8 million users, researchers were able to make a prediction with an accuracy of 64% to 87% using only the permission type and the app ID in a simple SVC algorithm [208]. Lastly, crowdsourcing can also be used to find an optimal tradeoff between denied permissions and utility of the app [165].

Location sharing

In the location sharing domain, a study by Iachello et al. [163] found that especially designing privacy management tools for location sharing has its own additional challenges, which they sum up in a design guideline. Automatic recommendation without the involvement of the user should not be done; it should be initiated *only* if the user actively requests it. Furthermore, such a privacy management system should support several functionalities particular to the location sharing domain, like the possibility to completely deny requests, return a wrong location, or a generic reply like "I am busy". Locations should be shared first only with individuals, before the location is shared with a group, to prevent sharing the location with a broader audience than needed. Lastly, the authors encourage use of a decentralized framework, so that the location data can be kept on the users' devices, and thus the need for security mechanisms on the provider's hardware can be limited. As a take-away message, the authors state that such new privacy systems need a longer training period than usual user interfaces, and that a steadily changing characterization of the users is needed to offer them suitable privacy settings, as the factors influencing

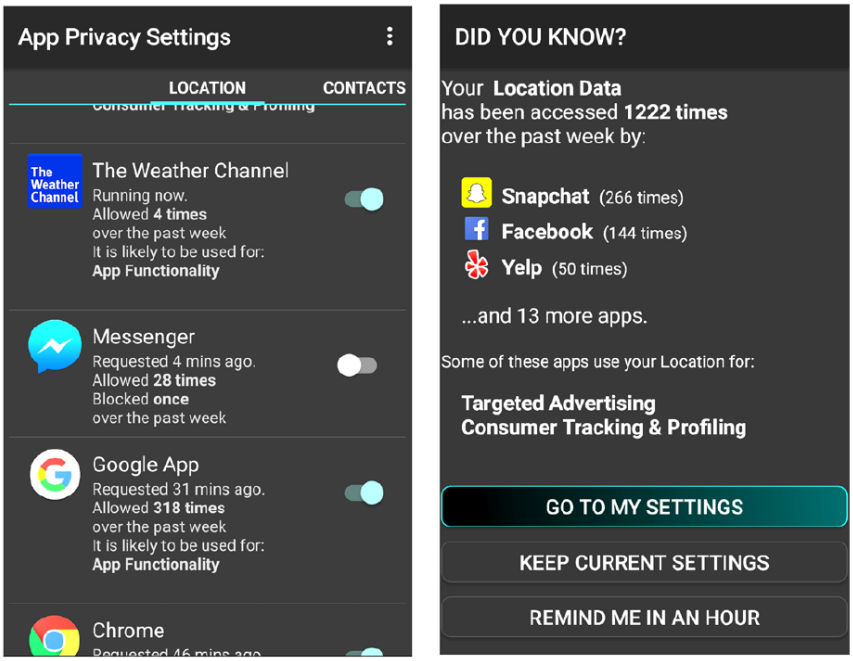


FIGURE 2.4: Modified permission screen (left) and detail screen (right) to notify the user about the frequency of access to personal data [209].

the privacy decision, like context factors, also steadily change throughout the users' lives.

Some of the privacy management systems make an automatic trade-off between privacy and security in location sharing, based on the possible attack scenarios that might occur depending on the privacy settings [298], taking the users and the attackers into account. Some of the approaches also take other stakeholders into account; these can provide security mechanisms but can also be the target of an attack. These include the social network providers, the infrastructure provider and also the government, which may record and pass the data to unwanted recipients either by mistake or as a victim of a hacking attack [288].

Similar to the aforementioned domains, most of the privacy management systems are rule-based and rely especially on context information when deciding whether to share a location or not [31, 321, 281]. Benisch and Kelley argue that the greater the complexity with which the rules can be defined in such a management system, the more this reduces the problem of undersharing, as users tend to disclose locations less than they would actually prefer (or not at all) if the privacy management system does not offer options for fine-grained privacy rules [31]. However, such a system involving complex rules only works if a sufficiently large number of rules is created, which is often not done by a typical user. In their management system, they use the time and place of the location as context factors, as well as the severity of a wrong disclosure, to adjust the privacy settings. Another system called "Super-Ego" takes another approach, and uses the privacy decisions of other users as a point of reference for the decision [321]. If the majority of other users share their location in the same context, the system decides to do so as well, unless the user explicitly decides differently. As context factors, they used the activity of the person, time and place, and the identity of the other users, as well as properties of the place like diversity of the audience, to make the prediction. Research has also discussed a way

to find persons while still maintaining their privacy, resulting in software called *Personfinder* [281]. Personfinder uses the time of day, day of the week as well as the location requester to decide on the disclosement of the location. In a study, the correctness of their disclosement was at 65% at the beginning, but could be improved using machine learning and a condition-based reasoner to about 79% in a follow-up study in the wild. Finally, if the location is shared on a mobile phone, the type of app can also be a context factor that is important for the sharing decision. Whereas marketing apps should typically not receive the location at all, some apps like a weather app still have the same utility if the are only given the current city rather than the exact GPS position. Other apps like sport tracking apps need the exact GPS position only to derive the pace of the runner, and can therefore also be given a synthetic route with the same speed profile [106]. Consideration has also been given to the interaction to share a location, especially as changes in the location privacy settings typically occur when the user is on the go, meeting up with other people, and does not have much time to open a complex privacy interface and change the settings [169]. One solution for this problem was proposed by Jedrzejczyk et al. in their "Privacy-Shake" approach, where the user can control the privacy setting using shake gestures with her smartphone [169]. However, although the approach was found to be promising, users struggled to do the gestures correctly, leading to a low success rate of only 40% and massive user frustration connected to this.

Privacy management from a social-driven perspective and in-situ feedback

Instead of thinking about location sharing privacy in terms of controlling data disclosement as a trade-off between privacy and utility, the problem can also be seen from a social-driven perspective [318]. Usually, people have a certain goal when sharing their location, which is more than just showing others where they should meet up. Often, sharing location is a type of impression-management, to shape the image of a person within their social circle. Therefore, the authors state that, according to their study results, users should share the activity they are currently doing more than the actual place, which is unimportant for impressing others. Instead of location obfuscation, ambiguous names should be used, as the user's actual friends will know which location is meant. Using these rules, it was significantly harder for a stranger to determine the actual location, even if she had access to background information like a map or typical routes of the user. In a study, the rate of correctly identified locations could be reduced to 50% compared to 91% for participants who were not instructed.

Also, in-situ feedback from users can be used to refine privacy settings, for example based on how many other users allowed the access in the past [131], or by actively asking the user whether to block a location requester once an anomaly is detected, for example if the requester sent a high amount of location requests in a short period of time [168].

The necessary preparatory work: grouping recipients

As we have seen in the aforementioned approaches, a lot of them define privacy settings based on the group of recipients or the requester's membership in such a group. However, we also discussed that users typically do not create such groups or "friend lists" due to the effort required. Research has tried to address this problem using different approaches, e.g. by using machine learning techniques and also involving the user in the grouping task. In the simplest form, friend grouping, a.k.a.

community detection, works on the social graph of a user [235]. The algorithm consecutively removes edges where the edge density is lower compared to the rest of the graph. By that means, the algorithm iteratively creates clusters which are unconnected to each other, forming the communities that are returned by the algorithm [235]. This approach, or manual friend grouping, works well for smaller groups which are highly connected or where the user remembers the names of their friends; however, this simple approach fails especially in detecting large groups [13]. Newer approaches take the user into account, and work especially well for large groups. Regroup [13] for example, and a similar work by Fang and LeFevre [103], use machine learning to actively support the user while he is doing the friend grouping: based on the friends that have been selected as group members so far, the algorithm derives similar users that might belong to the same social circle, and proposes to add them as well. Another research idea is to support the user in choosing the audience when a new post is to be published, rather than doing the friend grouping without a specific reason, helping the user to find out what meaningful groups could be given the example of the current post. Similar to the aforementioned approach, the system observes the recipients added so far, and proposes further friends or friend groups to be added to the groups of recipients, recalibrating and improving the system after every new post.

Studies have found that the usability of the friend grouping user interfaces of social network sites is not the actual problem [177, 336]. It is, rather, the required mental effort that scares people away from doing the grouping task and leads them to censor their posts rather than doing narrowcasting with friend groups [177, 336]. Unfortunately, as stated in Section 2.1.2, automatically created friend groups rarely reflect the desired friend groups and are thus unsuitable for narrowcasting as well.

As we have seen, early privacy management systems were based on rule systems. However, research has shown that involving machine learning and context factors can significantly improve the system. Our approach also uses machine learning for prediction, but uses also *individual factors* to enhance the prediction precision as well. In the next section, we will take a closer look at which context factors are typically used in the different domains.

2.4 Context factors, individual factors, and how to capture them

As we have seen in the last section, context factors are an important source for predicting privacy settings, especially in the location sharing domain. Interestingly, it is very easy to change the user's mind using money [162]. Even privacy fundamentalists who claimed not to share any data at all will share their data if they are paid for it. Even more interestingly, they still share their data if they are notified about the negative consequences that can, and also negative consequences that definitely *will*, arise [162]. However, it is not the goal of this thesis to push users toward sharing their data; we therefore do not further observe this method of influencing sharing decisions. Apart from financial matters, the sharing decisions of other users also influence a person's privacy decision [245]. A study has shown that there is an effect on the privacy choice towards conformity with the choices of other users, meaning if a user is told that other users reveal more of their data, the user will share more data as well, and vice versa. However, although the study was able to show that this effect exists, the effect was only a small one. The actual type of information that has to be shared has a significantly larger effect on the decision [245]. Another important

context factor in general is the *age* of the post or the data that has been published [16]. The older the post, the less people want to share it, typically because their life and personality has changed and they cannot identify with the post as much as they did at the time of writing. Especially if people begin a new chapter in their life, e.g. graduating from school or university, getting a new partner or moving to a new location, they tend to hide or delete old posts that do not fit their new life [16]. Other studies have found that, in general, the activity the user is currently doing, as well as the professional context, play an important role, e.g. whether the user is in leisure or working time, or is visiting an after-work event hosted by her company, which can be seen as half leisure, half working time [284]. Typically, the requester, the purpose for the request and the requester's friend group memberships also have an influence on the decision [153]. However, it is controversial whether the situation the user is currently in, e.g. whether she is currently at work, attending a concert, or playing a sport, has a large influence [199]. Some studies state that the situation and its attributes, e.g. whether it is a family party or a party with university friends, whether it is a birthday or farewell party, has a strong influence on the decision [153], whereas others came to the conclusion that the situation in general plays a role, but only a small one compared to the requester of the information [199].

2.4.1 Location sharing

Apart from the general context factors described below that apply to an online context like ubiquitous environments and social media, location sharing has been a special focus of research, as it has been shown that sharing a location significantly depends on the context [246]. Above all, the requester has been shown as the major factor of influence when sharing a location in several publications [246, 321, 74]. Apart from this, the location itself also plays a role, whereas researchers have a different notion of what is important within the location. Some state that the actual position is important [246, 31] whereas others say it is more the activity that is performed at the location [73, 321] or the type of people that typically visit such a place [321]. A third notion is that the professional context is the important factor regarding the location, e.g. whether it is a work environment, or a public or private space [74]. Finally, some even say that the location, activity or the situation plays only a minor role compared to the requester of the location [115]. Another factor that is also sometimes mentioned is the *purpose* of the location request, e.g. whether it is for meeting up with a friend, a family member wanting to feel safe about loved ones, or only an acquaintance who does not play an important role in the location sharer's life [246, 325, 74]. The fourth factor that was sometimes mentioned is the time and day of the week when the location was requested [321, 31], whereas newer publications came to the conclusion that it is more the activity and not the time and day that is important [246]. Lastly, there were also context factors that have been found to be important by one publication, but which have neither been substantiated nor disproved by others, like the frequency and granularity of the request [246], the benefit of the location sharing and the likelihood that the benefit is actually achieved [325], and even the mood of the location sharer [74]. Finally, if the requester is a company rather than a person, it can also depend on the type of company, whether the location request is accepted or rejected: Most people are fine sharing their location in public if the data is anonymized, an equal number of users allow sharing the data for research purposes only, others allow it for both research and commercial purposes, and others will share it with nobody. Finally, sharing the location with everybody without anonymization or only with commercial companies, is very rare [51].

2.4.2 Individual factors

Apart from those *context factors* which are independent from the actual user, there are several factors that depend on the user, like demographic factors, personality and also privacy desire [185, 359], which we combine under the term *individual factors*. Several publications have shown that demographic factors have an influence on the privacy concern of a user and thereby also on the privacy settings [185, 202, 212]. The demographic factor that was mentioned most frequently is the age of the user [277, 185, 212, 149, 82, 202, 238]. Interestingly, earlier studies denoting age as one of the influential factors found that privacy concern increases with an increasing age, meaning the youngest users have the lowest level of privacy concern [185, 82, 202]. However, newer studies show that this trend did not continue for the next generation: Most recent studies show the privacy concern decreasing up to the group of participants aged between 25 and 35 years; but younger participants (between 16 and 25 years) again have a higher privacy concern, indicating that privacy will again gain importance in younger generations [212, 149]. Another factor that has been found to be important in several studies is the gender of the user [277, 212, 292, 144]. In all studies that found a gender effect on privacy concerns, females had a significantly higher level of privacy concern [277, 212, 292, 144]. Interestingly, other studies claim that demographic properties like age, gender, nationality, marital status, employment status and income have no effect on privacy concerns [349, 251]; it is rather the risk that is perceived that a privacy violation will occur, as well as the potential damage associated with it [349]. The results for the influence of a user's education level are also mixed: Where some claim that people with a low level of privacy concern are usually poor and less educated [82, 359], others say that especially highly educated people have lower privacy concerns [251]. Similarly, people with a higher income seem to have lower privacy concerns [238]; persons with a higher position at their workplace tend to put more emphasis on security, rather than privacy of their employees [140]. Also, the nationality and the culture of the users seems to have an impact: In a study comparing the privacy concerns between US and Italian citizens, researchers found that Italian citizens are in general less concerned about privacy, demand less control from their government and also dislike privacy intrusion by the government more than US citizens [90, 91].

Apart from the demographic factors, the personality of a user, especially as represented by the personality measures of the big five personal inventory [78], has a significant effect on the privacy settings [24, 144], how they can be motivated in sharing data [213], and also how privacy user interfaces should be designed [145]. Persons with a high conscientiousness or emotional stability are likely to be significantly more concerned about privacy [24]. Also whether the user is female, and whether she has been target of a privacy invasion before, positively correlates with privacy concerns [24]. Depending on the personality, users are more or less susceptible to certain social engineering attacks. Users with a low neuroticism, for example, have been found to be more susceptible to phishing emails, where the recipient appears to have won a prize [144]. Openness and extraversion correlate positively with the amount of information shared on social networks [144]. Both introverted and extraverted persons can be motivated in sharing personal data by offering social benefit adjustments in return, which can be, for example, the ability to join a special club, or to be part of a community with special interests that fit the user's interests. Extraverted persons are motivated by both online and offline adjustments, whereas introverted people can be motivated only with offline adjustments [213]. Also, the user's self-concept makes a difference in privacy concerns. People with

an interdependent self, e.g. people that define their self through their relations with others, prefer proxy privacy control by industrial companies, whereas those with an independent self, who see themselves as unique individuals with their own characteristics, prefer technology-based privacy controls [352]. Even the day of birth and the corresponding zodiac sign significantly influences personality, according to a study [324]. Lastly, studies have shown that if the privacy UI is tailored towards the user's personality, this increases the privacy and security of the user's personal data [145].

In addition to the demographic factors mentioned, expertise in the field of privacy typically leads to an increased level of privacy concern [185]. People who follow privacy issue developments are often individuals with a high social awareness [88]. However, in contrast to expertise in the area of privacy, general internet usage experience does not influence privacy concerns [359]. The technology that is used to record the data can also play a role. A study has shown that users have a higher level of privacy concern when sensitive data is recorded using a Google Glass compared to a smart home system [212].

As we can see, there is mixed evidence on demographic factors and their correlation to privacy concerns; some say that they have an effect in one direction, other studies claim the complete opposite, whereas again other researchers come to the conclusion that demographic factors do not have any effect on privacy concerns at all. However, the negative correlation of some of the personality traits (like awareness and extraversion) with privacy concerns, user behavior and vulnerability to certain attacks seems to be uncontroversial in research [144]. We therefore decided to include the privacy concerns as well as the personality traits in our set of individual features, which are then used for predicting privacy settings. In the next section we will discuss in more detail how these individual measures are collected.

2.4.3 Personality and privacy questionnaires

Privacy questionnaires

Privacy questionnaires that have been constructed in research over the years have rarely built upon existing previous work. Therefore, we will discuss the developed questionnaires one after another in order of publication date.

Alan Westin's work

The first attempts to capture privacy concerns were made by Alan Westin in the second half of the last century, resulting in a categorization of users into three privacy categories [195]: *privacy fundamentalists* who do not want to share any data at all, or as little as possible; on the opposite the *unconcerned*, who are eager to give away all their personal data without any privacy concerns; and the *privacy pragmatists* between those two categories, who always make a trade-off between privacy and utility when it comes to sharing sensitive data. Alan Westin proposed a three-item questionnaire that allows participants to be assigned to one of the three categories [195]. However, it has been shown that these three categories and the overly short questionnaire are too coarse-grained in order to do a meaningful prediction of the user's behavior in privacy scenarios [349]. Users from the three categories do not behave significantly differently [349].

Westin's categories were later extended into four categories by Sheehan et al., inclucding again the *unconcerned* user, the *circumspect* user who has some privacy concerns, the *wary* user with a higher level of privacy concern, and finally the *alarmed* user who is similar to Westin's privacy fundamentalist [293].

Concern for information privacy (CFIP)

After Westin's first approach, several privacy questionnaires emerged, which mostly have not been thoroughly tested and validated. In the ongoing section, we will therefore describe the most influential questionnaires that have been *scientifically validated*. For whatever reason, most of the questionnaires were constructed independent from prior work. One of the exceptions is the CFIP[2] questionnaire by Smith et al., which is the first approach after Westin's privacy questionnaire to capture privacy concerns in a detailed manner, offering several continuous privacy measures [303]. The CFIP consists of fifteen statements, which have to be evaluated by the user on a seven-point Likert scale. The statements contain abstract privacy scenarios, which do not involve a specific privacy domain (like online privacy) or specific companies. The authors provide a detailed guide on how to evaluate the questionnaire using a weighted average calculation, resulting in four different privacy measures: *collection*, denoting how important it is for the user to know *which* data is collected; *errors*, meaning the concern that protections against errors in personal data are inadequate; *(unauthorized) secondary use*, describing the concern that data is used for another purpose than officially communicated; and *improper access*, measuring the user's concern that personal data can be accessed by unauthorized entities [303]. The CFIP has been proven to be distinct and reliable by later studies [314].

Sheehan's work

A later approach by Sheehan et al. is based on the *fair information principles* previously published by the US Federal Trade Commission [3]. In contrast to the CFIP, this questionnaire used realistic scenarios instead of abstract questions in order to receive feedback from the user [294]. More specifically, the questionnaire consisted of 14 scenarios in which a privacy violation took place, mostly scenarios around emails and websites. The authors pointed out three major privacy factors which are important: the control over collection and usage of information (similar to the *collection* measure of the CFIP), and whether it is a short-term, transactional relationship, that is established only once for a specific transaction like an online purchase or data acquisition in a questionnaire in return for a refund, or a long-term relationship with a company that has been established for some time and that is part of an ongoing relationship with the company [294].

[2] concern for information privacy

[3] http://www.lawpublish.com/ftc-fair-information-practice-principles.html (last accessed: 2020-03-09)

Internet User's Information Privacy Concerns (IUIPC)

Based on the CFIP, researchers later developed the Internet User's Information Privacy Concerns (IUIPC) scale [221]. Being the first to take advantage of earlier findings and questionnaires, they reused some parts of the CFIP while making the abstract questions more concrete, by rephrasing them to fit the internet context (for example by using "online companies" instead of "companies"), especially the context of online shopping. Furthermore, they added new questions, for example regarding the user's privacy practices. The questionnaire allows one to compute three distinct privacy measures, namely *collection* similar to the CFIP, the *awareness* of what happens with the private data, and *control*, denoting how important it is for the user to have control over the flow of her private data, by whom it can be accessed, how it is stored, and for what purpose it is analyzed [221].

Privacy Concern Scale (PCS)

Another questionnaire that is focused on online privacy is the Privacy Concern Scale (PCS) [52]. In contrast to the IUIPC, the PCS has a broader notion of privacy which is *not* limited to data privacy. More specifically, it allows one to describe the user's privacy concerns in three dimensions: first, the *general caution* with sensitive data in an online context, similar to the IUIPC; second, the usage of *technical protection* mechanisms for data privacy like pop-up blockers, private browsing and deleting the browser cache; and lastly, the *privacy concerns* against other entities, like persons or companies not being who they claim they are, or forwarding an email inappropriately. Instead of letting the user rate privacy scenarios, the questions directly ask about the user's behavior and concerns. The measure *general caution* has been shown to correlate positively with the IUIPC measures [52]. Also, the Need for Privacy Questionnaire (NFP-Q) uses such a broad notion of privacy, by also capturing the user's need for physical privacy and interactional privacy, denoting the user's desires to have a safe haven and to be left alone [323].

Privacy questionnaire by Earp et al.

A longer privacy questionnaire has been published by Earp et al. [96], and uses 36 items to capture the privacy concerns and practices of a user regarding six different factors outlined by the authors, using scenario-based questions similar to the IUIPC. Those factors describe the user's desire for personalization, for example for targeted advertisements; the awareness of data sharing and processing similar to the IUIPC measures *awareness*; concern about unauthorized transfer of the data; knowledge about which data is collected and where it is stored; and control over the access regulations on the private data. Unfortunately, although the authors received good statistical results for their scale, they based their results only on a single sample of respondents; therefore the scale is not seen as reliable in research and thus is not used often [256]. Another attempt to capture privacy concerns in the context of web actions has been made by Dinev and Hart, and offers two measures to categorize a user, denoting how seriously data misuse is perceived by the user, and how grievous it is when a third party can find out information about the user in question [89].

Indirectly capturing privacy concerns

The aforementioned privacy questionnaires all directly asked the user to rate either a scenario, or how great the user's privacy concern is in different subject areas. Unfortunately, capturing privacy concerns this way itself influences the user's privacy concerns, depending on the framing of the privacy questions. When privacy is made salient, this increases the user's level of privacy concern in the responses [48]. Braunstein and Granka therefore propose to derive privacy concerns indirectly, and offer a data-centric questionnaire which asks about the user's concern toward losing data. Although the approach looks promising, they did not conduct a professional statistical analysis and did not perform a factor analysis; therefore the method cannot be seen as reliable [48]. Finally, there also exists a privacy scale which is tailored especially towards privacy in the context of social media, observing how the social interconnectedness corresponds to the user's privacy desires in a social network [347]. This questionnaire consisted of four different blocks of questions, asking about Facebook usage intensity and social connectedness on Facebook, which they tried to match with bridging social capital, denoting how much Facebook is used to get in touch with people and get to know about new things, as well as bonding social capital, asking whether Facebook is used as a platform to talk about sensitive topics like personal problems with other users.

To conclude, there exist many privacy questionnaires which have mostly been generated without using insights from prior work. The CFIP can be seen as the most influential questionnaire, as it has been adopted for other questionnaires and is also used in research very frequently [256]. The IUIPC questionnaire is the second most used questionnaire [256] and is tailored towards online privacy, making it a good candidate for our research. Throughout this thesis, we will therefore use the IUIPC questionnaire for all domains, as it offers a generic view on the *online privacy concerns* of a user independent from the actual domain. In the social media domain, we will also include parts of the more specific social network privacy questionnaire by Wisniewski et al. [347] to show an example of how introducing a specialized privacy questionnaire can improve the user model and allow a higher prediction precision.

Personality questionnaires

Whereas the work in the field of privacy settings has mostly ignored prior contributions, it is just the opposite for the personality questionnaires. Throughout history, one can find three major development tracks for personality questionnaires, namely the works by Robert Cloninger resulting in the Temperament and Character Inventory (TCI) and its successors [72], the questionnaires by Hans J. Eysenck called Eysenck's Personality Inventory (EPI) [42] and later the Eysenck Personality Questionnaire (EPQ) [101], and finally the (most popular) big five personality traits [78].

Big five personality traits

The roots of capturing personality can be found at the beginning of the 20th century, where researchers started in 1936 to capture personality traits by sampling language, e.g. by analyzing and clustering adjectives that describe facets of a human's personality [7]. As a result, researchers came up with a list of 4504 personality adjectives

[7], later reduced to 171 items by eliminating duplicates, resulting in the sixteen factor personality model [21]. Research on capturing personality stalled in the middle of the 20th century, but came back into focus in the 1980s by the work of several researchers, resulting in the aforementioned personality questionnaires. The NEO-I questionnaire from 1976 is said to be the first predecessor of the big five personality traits [76]. Its authors Costa and McCrae again went for a cluster analysis, which led them to a questionnaire containing *three* personality traits, namely *extraversion*, *neuroticism*, and *openness to experience* [76]. The NEO-I was later refined and extended with the two personality traits *agreeableness* and *conscientiousness* using the study data from the Augmented Baltimore Longitudinal Study of Aging [297]. The new questionnaire included six facets for the three personality traits of the original NEO-I and was named NEO-PI by the authors [77]. Also, the NEO-PI was revised by the authors to have six facets for each of the five personality traits, resulting in the NEO-PI-R [78], which is still used as the current standard for capturing the big five personality traits. The five-factor model was later supported by an approach based on lexical analysis, which also emphasized the five mentioned personality traits, denoting them as the "big five personality traits" [133]. The NEO-PI-R takes about 45 to 60 minutes to complete; therefore, research created an alternative shorter version called the NEO five-factor inventory (NEO-FFI), which takes only ten to fifteen minutes to complete [45]. There also exists a version called "Ten Items Personal Inventory" (TIPI) which is meant for a "quick and dirty" evaluation of the big five personality traits, requiring only ten items and less than one minute to complete [137]. The NEO-PI-R is currently established as the standard for capturing personality and has also received good critiques from other researchers [172], although there also exist some negative aspects that are not solved by the questionnaire, like the social desirability bias which leads users to be dishonest about their personality and therefore the answers to the questionnaire [39]. It has also been shown that the big five personality traits can be predicted using a person's writing style, for example on social media websites or in online blogs [104].

Eysenck's personality questionnaires

Eysenck's personality questionnaires were also developed throughout the 1980s, starting with the Eysenck's Personality Inventory (EPI) [42] containing 57 questions to be answered "yes" or "no", and resulting in two personality dimensions, called extraversion/introversion and neuroticism/stability, similar to the corresponding big five personality traits. Based on this work, the authors established the Eysenck Personality Questionnaire (EPQ) [101], having a total of 57 questions with the same binary scale, to offer two additional personality dimensions, namely *psychotisism/-socialisation* meaning how aggressive, egocentric or manipulative the person is, and *lie/social desirability* stating whether a person is willing to lie to be accepted by other or to conform to social norms. This questionnaire was later extended to the EPQ-R using either 100 questions for the same personality dimensions, or 48 in the short version [102], and was finally transformed into the Eysenck personality profiler, which also offers facets of the mentioned personality dimensions [344].

Robert Cloninger's work

Parallel to that, Cloninger started his research in the early 1980s, resulting in his first attempt toward a personality questionnaire called the "tridimensional questionnaire" [71]. As the name suggests, the questionnaire captures personality in three dimensions, denoting how novelty-seeking the person is, how important it is for them to avoid personal harm (expressed by shyness and anticipatory worry, for example), and the reward dependence of the person. Based on this 100-item questionnaire which consisted of "yes"-or-"no" questions, the authors developed the Temperament and Character Inventory (TCI) [72], which used 240 items on the same scale to calculate the three aforementioned personality measures which are now called the "temperaments", and added a new temperament as well as three "characters" to their list of output variables. The fourth temperament is called *persistence*, meaning how perfectionist and hard-working the person is. The three characters denoted the *cooperativeness* of the user; the *self-directedness*, meaning how responsible and purposeful the person is; and finally the *self-transcendence*, denoting the spirituality of the person [72]. The scale was finally revised to the TCI-R scale, using the same questions but with a five-point rating instead of a simple yes/no choice. Later studies have shown that the TCI-R has a close relation to the personality questonnaires by Eysenck [358] as well as the big five personality traits [122]: Harm avoidance is positively correlated to neuroticism and negatively to extraversion, novelty seeking is correlated with extraversion, persistence with conscientiousness, and cooperativeness with agreeableness, to mention only some of the correlations.

To conclude, there were several lines of development of personality questionnaires. However, research has shown that the questionnaires correlate to each other; it is therefore likely that all of them are capturing the same results, only in a different form or with a different labeling of the personality traits [358, 122]. For our research, we therefore concentrate on the current scientific standard for characterizing personality, namely the big five personality traits.

2.5 Privacy User Interfaces

One of the biggest problems regarding privacy is that users are typically not aware of either the negative consequences that can occur when over- or undersharing data, or the mistakes in their own privacy settings that can lead to these consequences [214]. Also, pop-up warning messages are often not seen, or ignored [214]. Privacy awareness and transparency of the privacy settings has been shown to increase the notion of privacy risks, and is therefore essential and has been a central part of privacy user interface research in the past [214]. It has been shown that a lack of transparency is an important aspect also in privacy recommender systems [301]. Typically, a "black box" recommender system that does not allow the user to understand why they are receiving certain recommendations often leads to a reduced trust towards the system [154]. Previous work on privacy user interfaces can be divided into five different kinds of approaches, which will be discussed in the remainder of this section:

1. **audience selection user interfaces** that are either based on *interpersonal distance* or a *radar* metaphor to let the user select the right audience while having an overview over all possible recipients;
2. **reducing the complexity** of the privacy settings by using abstractions or hierarchical visualizations;
3. **visualizing information flow** of the user's private data to other users and thereby increasing privacy awareness;
4. **grouping user interfaces** which are needed to create friend groups used for *narrowcasting*; and
5. **consequence-based privacy** user interfaces that increase the user's privacy awareness by emphasizing the possible consequences arising from their privacy behavior.

2.5.1 Privacy setting and audience selection tools

Instead of visualizing already existing data flows and letting users correct existing privacy settings, another approach is to limit the data flow when the data is disclosed, for example by limiting the audience of a newly created social network post. Social networks typically offer list-based UIs for that purpose. Figure 2.5 shows the custom privacy setting dialogue of Facebook, where the user can type in names of friends or friend groups that should be able to see the post (upper text field) or that should be excluded from the audience (lower text field). When the user starts typing, the website provides an auto-complete functionality and displays friends or friend groups matching the entered text. However, the user still has to remember the names of the friends to include or exclude, and cannot see the complete current audience of the post. Research has begun to enhance such privacy setting dialogues using two different kinds of approaches. One uses the *interpersonal distance* between the user and her friends to align the audience in the UI; others use a *radar metaphor* to arrange privacy choices in a clustered way.

Studies from the first day of Google+ have found that users select the recipients of their posts according to the tie strength between the user and the friend as well as the aspects of their life they belong to [177]. Although the result is not perfect and will not completely reflect the tie strength of a user and his friends in real life, the tie strength can be approximated using data available on a social network, like

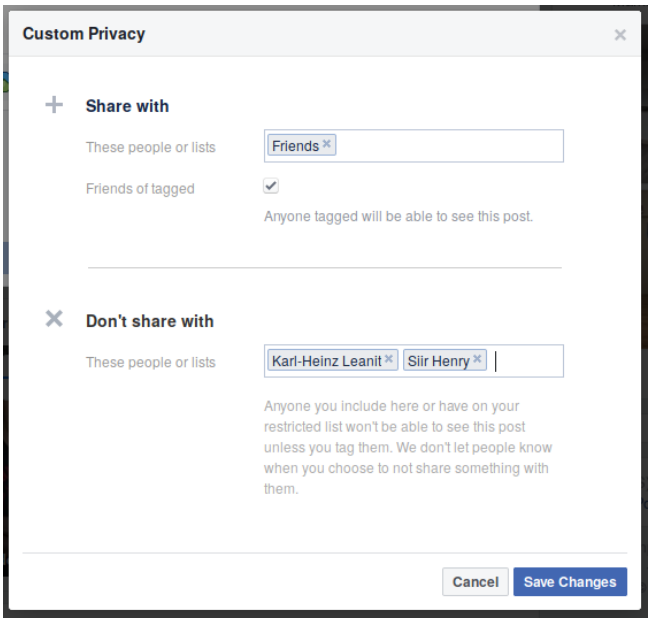


FIGURE 2.5: Custom privacy settings dialogue of Facebook.

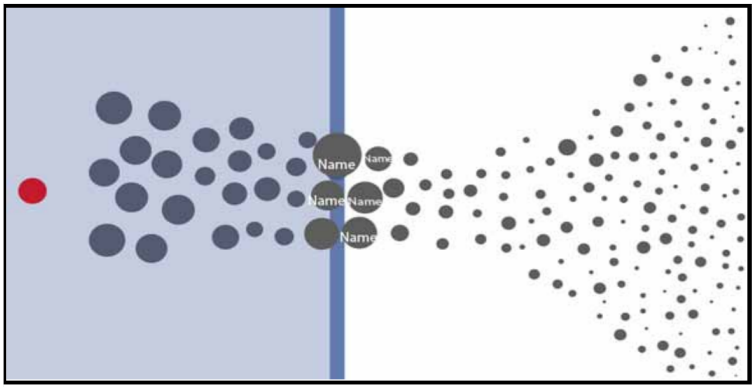


FIGURE 2.6: Selecting the post audience in a UI based on interpersonal distance as a criterion.

amount of likes, chat messages, comments, profile views and time since the friend was added [351]. Based on this tie strength calculation, Kauer et al. designed a tool for selecting the Facebook audience based on the interpersonal distance [180]. Their interface concept (Figure 2.6) was called a *slider* and aligns the friends from left to right with descending tie strength and decreasing size of the displayed profile picture and name, so that the friends with the highest tie strength are given the most space in the UI. Using the blue slider, the user can select the audience, including all friends with a tie strength up to this point (area highlighted in blue). Although the initial concept treated all friends the same way and aligned them all in one slider, the idea can also be expanded to support selection according to the tie strength in the different friend groups the user has created, by displaying one slider for each friend group on top of each other [180]. In an evaluation with a paper prototype, the authors were able to show that the amount of unwanted disclosures can be significantly reduced with the new UI (239 errors with a standard interface compared to 92 errors with the new concept).

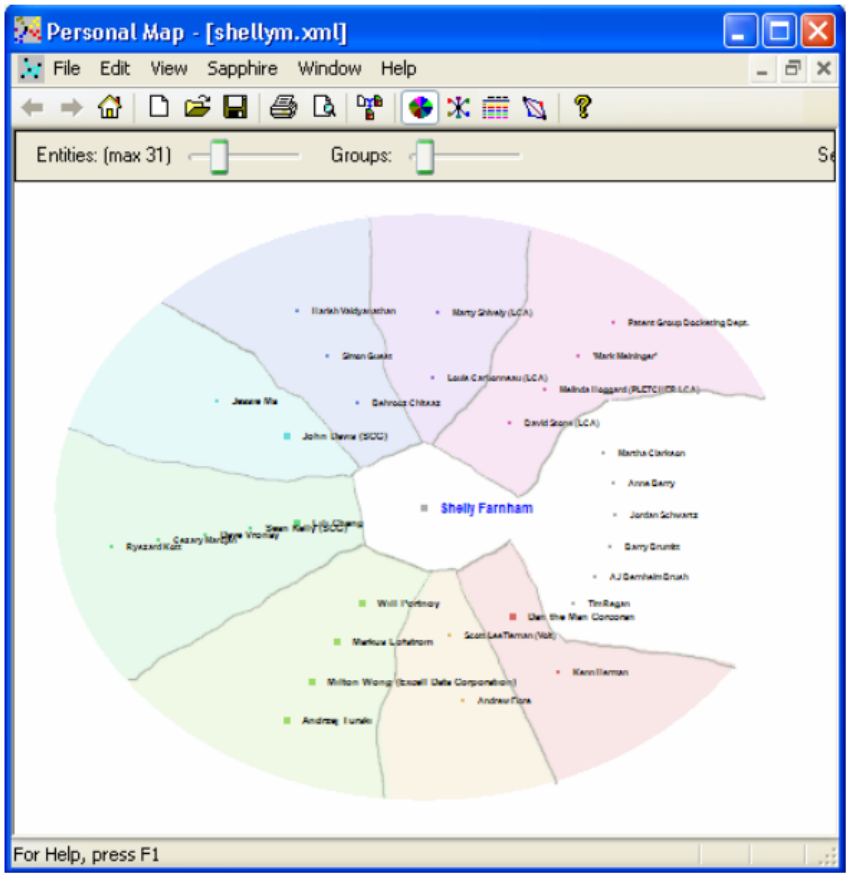


FIGURE 2.7: PersonalMap user interface for selecting email recipients.

PersonalMap is a similar approach, which allows the user to get a better overview of her most important e-mail recipients, as well as clusters inside the recipients, e.g. recipients which are often emailed together (like workcolleagues or friends from a sports club) [105]. In their UI (Figure 2.7), the user is placed in the center, and the friends around her. The more important a friend is, the closer she is placed to the center, near the user. In addition to this, the friends are also clustered according to their tendency to be emailed together, forming the groups denoted by the colored areas in the UI. PersonalMap received good usability scores and was rated to be highly accurate by the users; however, the participants in the user study stated they would most likely not use the tool in their daily life, possibly because the tool lacks integration into an email client.

The second type of interfaces for defining privacy settings that can also be used for other privacy tasks than selecting the post audience is based on a radar metaphor. Typical user interfaces for privacy settings (see Figure 2.8 for the Facebook general privacy settings page as an example) stack the privacy settings on top of each other, offering a slider or a link to change the setting for each of the entries, resulting in endless lists of privacy options, which are divided into several sections, and often scattered all around the website. Instead of this list-based metaphor, research successfully tested the radar metaphor to enhance the privacy overview and thereby also users' motivation to check and adapt privacy settings [67].

The most prominent example of such a privacy radar might be the privacy radar by Reinhardt et al. [69]. Their UI (see Figure 2.9) allows the user to define the privacy

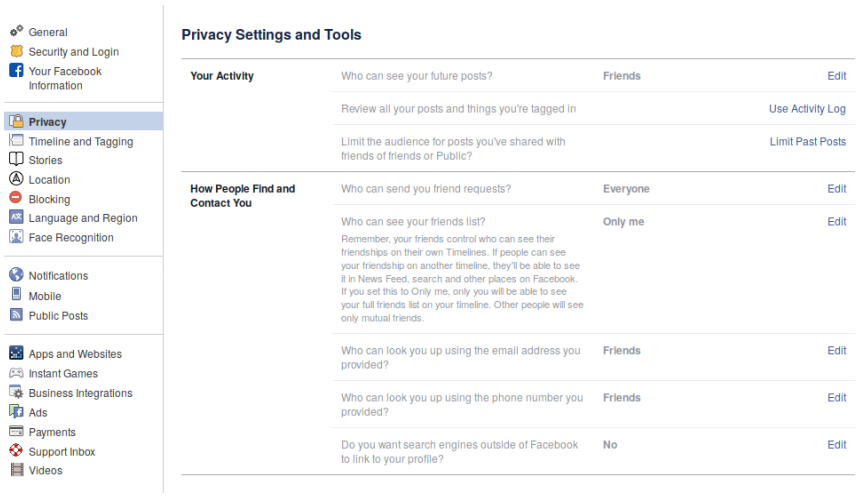


FIGURE 2.8: General privacy settings page of Facebook.

settings for data generated by the user's smartphone, using a radar metaphor: Each corner of the UI is dedicated to one type of sensitive data recorded by the smartphone, namely images, location data, audio, and accelerometer data. Each data type has several abstraction methods, allowing the user to share e.g. only the images without faces, or only the street level of the current location, denoted by the dots that are aligned from the respective corner to the center of the UI. The closer to the center, the more critical and the more sensitive is the data that is shared, also denoted by the colors of the dots from red (highly sensitive, unmodified data) to green (highly abstracted, less critical data). The user can select the corresponding privacy level by clicking on the corresponding dot. The UI also records which apps have had access to the mentioned data items, and offers the user an overview on this access history. If an unsafe setting is detected (like the unmodified audio in Figure 2.9), the UI warns the user about possible consequences using case examples. Later studies have shown that besides having a significantly better usability and providing a better overview than a list-based UI, it is also easier for a user to spot critical settings, which motivates the user to review and adapt the privacy settings [67].

A similar design was implemented by the Privacy Badge, which had the goal to increase the privacy awareness especially on small devices [129]. The user is placed in the middle of the radar screen, which consists of several layers. If a data item is shared, a symbol appears on the radar denoting the shared data type (for example a "$" sign for payment information, or a "+" denoting location information), together with information about the time and date of the data request, as well as the service provider requesting the data (for example "amazon.com" or "map24.de"). The closer the data point is to the center (the user), the more sensitive is the shared data type. The authors evaluated the design concept and the functionalities of the PrivacyBadge, which were rated positively overall. However, a study comparing PrivacyBadge to the current standard or an in-the-wild study remained for future work [129].

Apart from privacy settings, the radar metaphor has also been used for other purposes, for example for expressing the desired music style for music recommendations [227], as shown in Figure 2.11. The interface supports five different acoustic properties whose importance for the recommendation can be specified by the user,

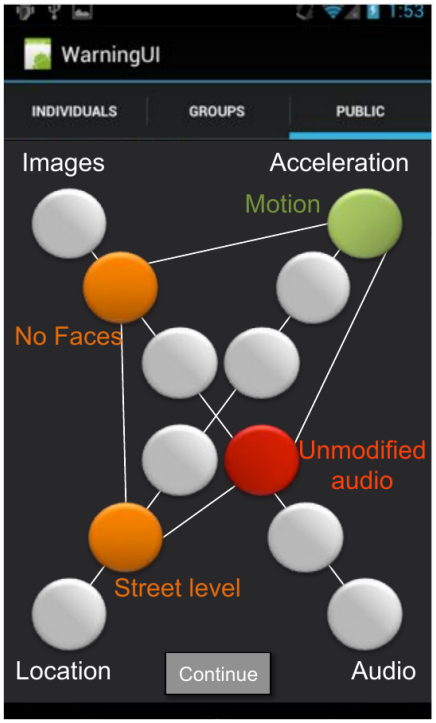


FIGURE 2.9: Privacy radar for selecting the privacy settings for private data on mobile phones.

FIGURE 2.10: Privacy Badge showing which sensitive data has been shared and with which service provider.

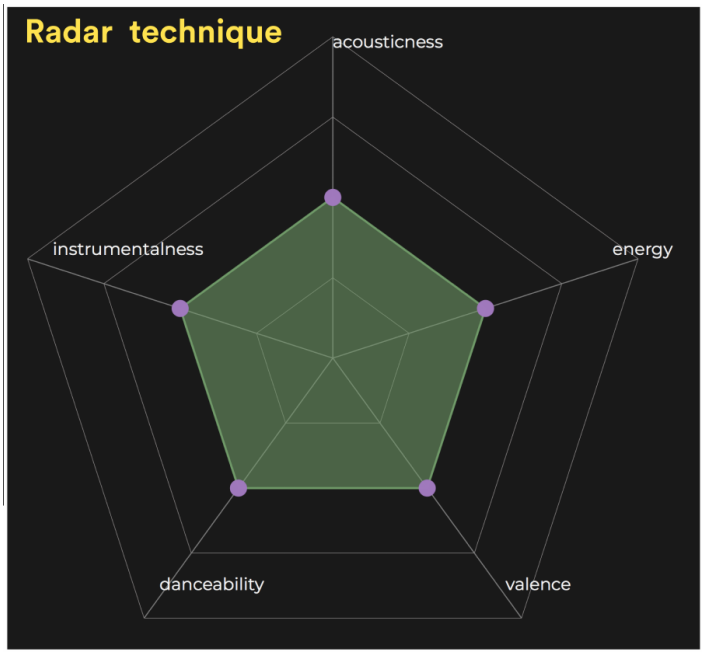


FIGURE 2.11: Radar interface for controlling several musical aspects for music recommendations on spotify.

using either the shown radar interface or a conventional slider interface. By grabbing the dots at each edge of the web diagram, the user can change the importance of the corresponding acoustic property to be weighted as more (drag towards the *outer* rim of the web) or less (drag towards the *inner* rim of the web) important. Also here, the authors were able to show the interface based on the radar metaphor was superior in terms of usability. Apart from these traditional two-dimensional interfaces, the usage of new visualization concepts like virtual reality (VR) can also increase the usability of an interface [170, 302]. Virtual reality gives users the feeling of really being in the virtual world, leading to a more natural and efficient interaction that would not be possible within traditional 2D or 3D applications [170, 302]. Research has shown that using VR in the e-commerce domain, for example for online shopping, is perceived as more useful, immersive and interesting compared to current two-dimensional user interfaces [201, 309].

2.5.2 Reducing the complexity of privacy visualizations

For *reducing the complexity* of user interfaces and reducing the space needed to visualize large data structures (such as privacy settings), the visualization community has focused on *mapping-based* or *clustering-based* approaches in the past: Mapping algorithms, for example principal component analysis (PCA), apply dimensionality reduction techniques to map the high-dimensional data space into a low dimensional space suitable for a visualization [254], e.g. by projecting thumbnail images to coordinates in the user interface according to their feature vector. On the other hand, clustering-based approaches, for example multi-dimensional scaling (MDS), try to group together similar items, like similar thumbnail images, so that the UI can choose and display only one representative of the cluster instead of all of them together [254]. The user can then browse through the data set by first selecting the representative, and then browsing through the contents of the selected cluster [253].

A clustering approach that is specialized towards clustering and sorting images is content-based image retrieval (CBIR), using features like color, shape or texture of the image for clustering [83]. A third approach for visualizing larger hierarchies is to use a so-called *fisheye technique*, which can magnify the area the user is currently interested in, allowing more space to be dedicated to the area of interest while compressing uninteresting parts of the hierarchy, allowing the user to keep a view of the whole hierarchy while still being able to see the details needed at the moment [196]. Another possibility for increasing the visibility of complex data structures is to transform the two-dimensional visualization into a three-dimensional space [257], for example by using the FADE algorithm [258] or recursive hierarchical space decomposition of the graph nodes [257].

In privacy user interfaces, researchers further reduced the complexity of the user interfaces, so that the current privacy status can be seen at a glance [182]. Kelley et al. for example used the metaphor of a nutrition label to allow users to review and especially to compare the privacy standards of different websites at a glance [182]. The work is based on the data provided by the platform for privacy preferences project (P3P) [120], which was started by the world wide web consortium (W3C) with the goal to standardize website privacy policies into a machine-readable form, rather than informal text-based privacy policies. Based on this data set, earlier researchers implemented the *extendable grid* user interface for displaying the aforementioned data [273]. However, user studies have shown that the visualization is not comprehensible to users, as some of the labels are not clear to users, and there is redundant information, as well as some further problems regarding the general usability of the UI [273]. Kelley et al. therefore used general principles from the nutrition labeling literature to reduce the amount of information shown to the user, resulting in a nutrition label for privacy policies, as shown in Figure 2.12. The design has the layout of a table, where the types of information are shown in rows, and the way the information is used in columns. Light symbols indicate that data is not collected or that it is only collected when opting in, whereas critical items in the table, where data is definitely collected that way or unless the user opts out, are marked with a saturated label in shades of red. The labels of the rows and columns contain short descriptions; more detailed descriptions can be found on a linked Useful Terms page. In an evaluation study, the authors were able to show that their nutrition label allows users to better understand and compare the privacy policies of two websites compared to a privacy policy written in natural language. Furthermore, users were wuicker to understand the policy, and stated the information was easier to find within the nutrition label [182].

An even simpler privacy UI for P3P website privacy policies was published by Cranor et al. earlier [79]. Their *privacy bird* allowed users to specify their privacy preferences in a privacy dialog, and then notified the users whether a website matches their privacy preferences using a trayicon, as shown in Figure 2.13. A green, happy, singing bird (upper left) shows that the website's privacy policy matches the user's preferences, so no further action is required. If a red exclamation mark is shown, then the privacy bird did not find any item on the website that mismatched the user's preferences, but there is embedded content that doe not match the preferences or that do not have a P3P policy. The yellow bird means that the website does not have a P3P policy, whereas the angry shouting bird informs the user that one of the privacy preferences does not match. If privacy bird is turned off, a gray bird is shown. To reinforce the notifications, a sound is played corresponding to the new state whenever the state of the privacy bird changes. According to their study, the privacy bird made it easier and faster for the users to spot whether their privacy

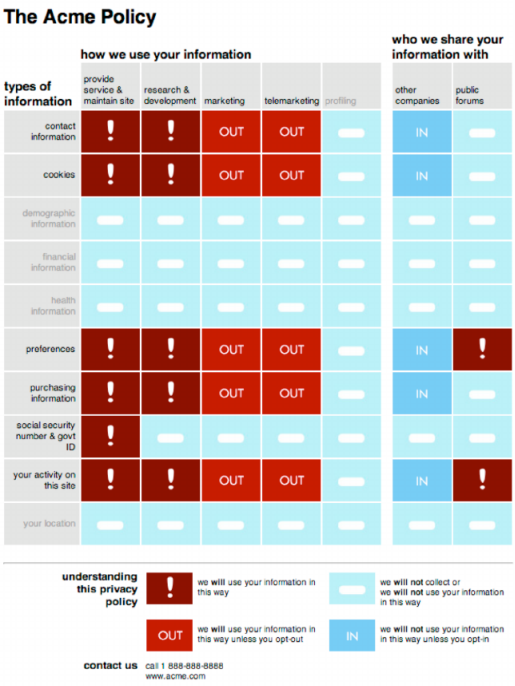


FIGURE 2.12: Privacy nutrition label by Kelley et al., allowing usersto see a website's privacy policy at one glance

FIGURE 2.13: Tray icons of the privacy bird, denoting whether the current website matches the user's privacy preferences

preferences are met, compared to reading the textual privacy policy or using the P3P visualization of Internet Explorer. Furthermore, the privacy bird was rated to be more useful, more helpful in understanding the privacy policy, and more likely to be used in the future than the privacy visualization of Internet Explorer [79].

Whereas the two aforementioned systems simplified the visualization and understanding of a privacy policy, others sought UIs that could display a privacy policy in an understandable way, and also allow the user to select appropriate privacy settings, while reducing the complexity of the display of information as well as the burden of choosing correct settings to a minimum [331, 22]. One such system is PRICON, which was created for a smart automotive environment, where the user has to choose which of the in-car data can be shared, and in what way [331]. Pricon reduces the complexity of privacy to four categories, *Anonymity*, *Data Sharing*, *Storage Time* and *Profiling*, allowing the users to easily compare the different privacy templates that are offered as choices (see Figure 2.14). The four categories were derived in a small-scale pre-study involving judicial privacy experts, IT experts and experts in human factors. Apart from the overview page to select the privacy template, the tool also offers more detailed information for each privacy template on a

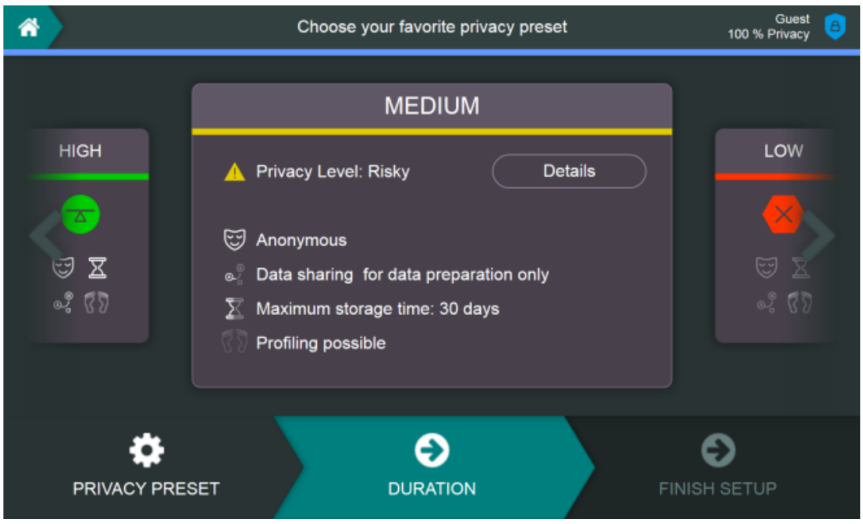


FIGURE 2.14: UI for selecting the privacy template in PRICON.

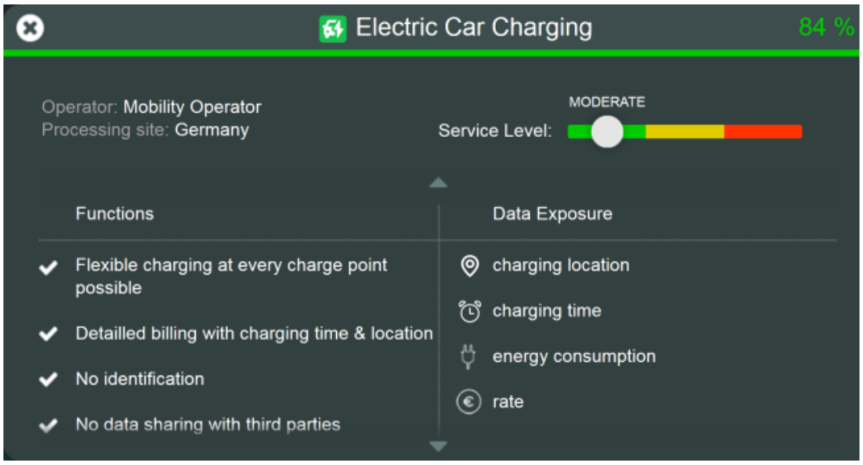


FIGURE 2.15: Detailed view for weighing up risks and benefits of each smart car service.

details page, and a services page weighing benefits and costs for each of the smart car services (Figure 2.15). The authors performed a user study to assess the usability of the UI, receiving an above-average SUS score of 75, and also an attrakdiff score that was slightly above average [331]. However, a comparison with traditional in-car user interfaces was not performed.

A third way of designing a privacy UI is through a data-driven approach, as has been done by Bahirat et al. when designing their privacy UI for choosing appropriate privacy settings in the IoT domain [22]. In a study, they first observed users' behaviors in privacy decisions, and which factors influence them. Their results showed that *who* is requesting the data has the highest effect, followed by *what* data is requested. The *reason* why it is collected (for health and safety, for example) had the third most importance for the decision, whereas the persistence, e.g. whether the data is only collected once or continuously, had the lowest level of influence. *Where* the data is collected did not have a significant effect on the privacy decision. Based on this hierarchy, the authors designed a hierarchical user interface that allows the users to tune their privacy settings, as shown in Figure 2.16.

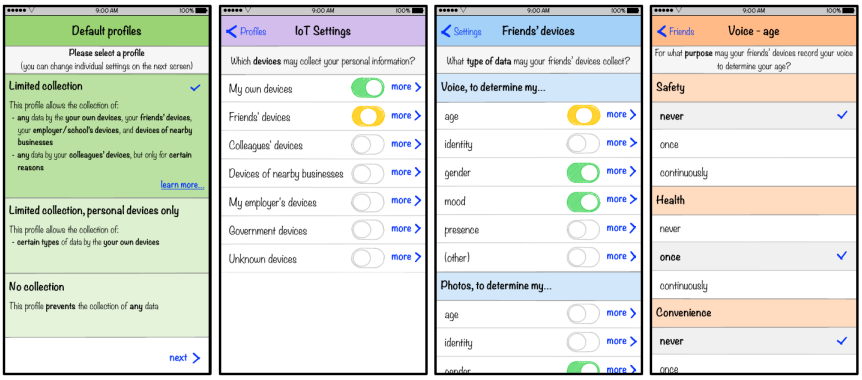


FIGURE 2.16: Layered privacy UI for IoT devices.

FIGURE 2.17: "Audience view", allowing a user to see which information other users can see when searching for his profile on Facebook.

The screen on the left-hand side is the initial screen, showing the available privacy profiles and allowing the user to select the desired one. When the user clicks the "learn more" link, the first layer of the privacy setting visualization is shown, listing the recipients (*"who"*) that can receive the data. When "more" is clicked, the second layer is shown, letting the user decide *what* data should be accessible. Finally, clicking again on "more" opens the third hierarchy, where the user can choose the reason and persistence level of the data disclosement.

2.5.3 Visualizing information flow to increase privacy awareness

Users often underestimate the number of recipients for their disclosed information, as well as the amount of information spreading associated with this [34]. Research has therefore tried to improve privacy awareness by offering users a graphical user interface that displays the spread of information in an easily understandable way. A very simple way of doing so is the *audience view*, which was also included on the Facebook website some years after it was discussed in research [207]. In such an audience view, the user can see how her profile looks when other users such as friends, other social network members or a public user visits the profile, i.e. which information the respective group of users can see from their own profile (Figure 2.17). A study has shown that using an audience view makes it significantly easier to find out whether another user can see personal information like one's name, email address, or birth date [207].

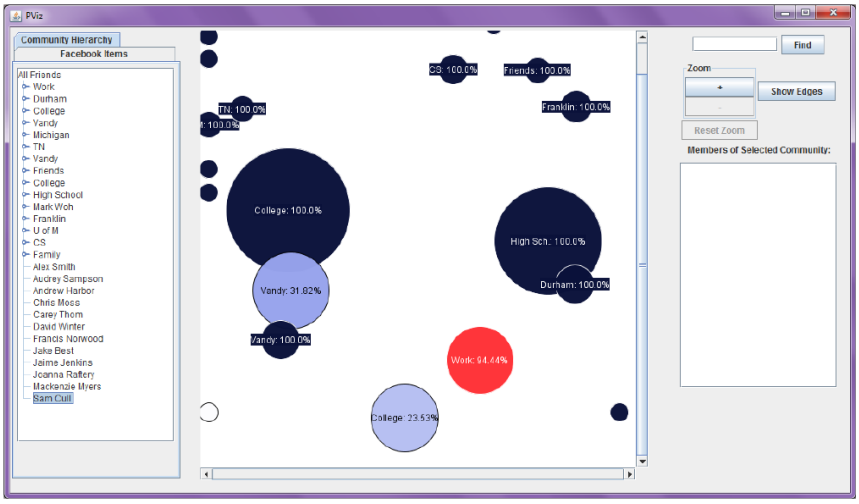


FIGURE 2.18: Friendship circles in PViz, allowing to hierarchically browse the friend lists.

Other works concentrated more on the display of a user's direct friends and also the friend graph connected to them. Early works used simple list-based interfaces enhanced by community detection algorithms [40], so that the friends can be sorted and displayed in multiple lists according to the detected communities [210]. PViz [224] and one of its successors called FreeBu [124] thereby concentrated on visualizing the user's friends and friend lists and their connections in an overseeable way. Initially, the tool partitions the user's friends into friend groups, as well as individuals who do not belong to any friend group, and visualizes them by circles (Figure 2.18). The privacy settings for the individuals in a group are denoted by the average percentage of visible profile items, ranging from 0% (no visibility) to 100% (all profile items visible). If an outlier is detected inside a group (e.g. a user who has privacy settings significantly different from those of the other group members), the circle is colored in red to highlight that a review of the privacy settings might be suitable for this friend list. When clicking on a circle, the tool opens the friend list and displays each contained friend as a circle in the visualization, again showing percentage values denoting the respective disclosement settings. Compared to the Facebook defaults, PViz has been shown to be as effective to detect the privacy settings for single users, but is significantly better for determining the privacy settings of a group of users.

Based on the work of FreeBu, researchers implemented other possible designs and evaluated them against the circle-based design used by FreeBu #1 and PViz [85]. Those encompass (from left to right in Figure 2.19) a *map-based* visualization, that displays the friends in a graph-based manner, where friends are nodes and vertices are the friendship connections between them; a *column visualization*, listing the friends in different columns according to the detected friend groups; and a mechanism to create friend groups based on the interpersonal distance, similar to the work by Kauer et al. described in the next subsection [180]. Each of the interfaces had its own advantages and drawbacks; only the column-based interface received lower usability scores. The study discovered that users typically only interact with their top 20 friends, thus making the rank-based interface most suitable of all those proposed.

Some of the friendship visualizations go even one step further, visualizing not only the flow to the direct friends, but also to the friends of the friends and the

FIGURE 2.19: Alternative friendship visualization techniques as implemented by FreeBu 2.

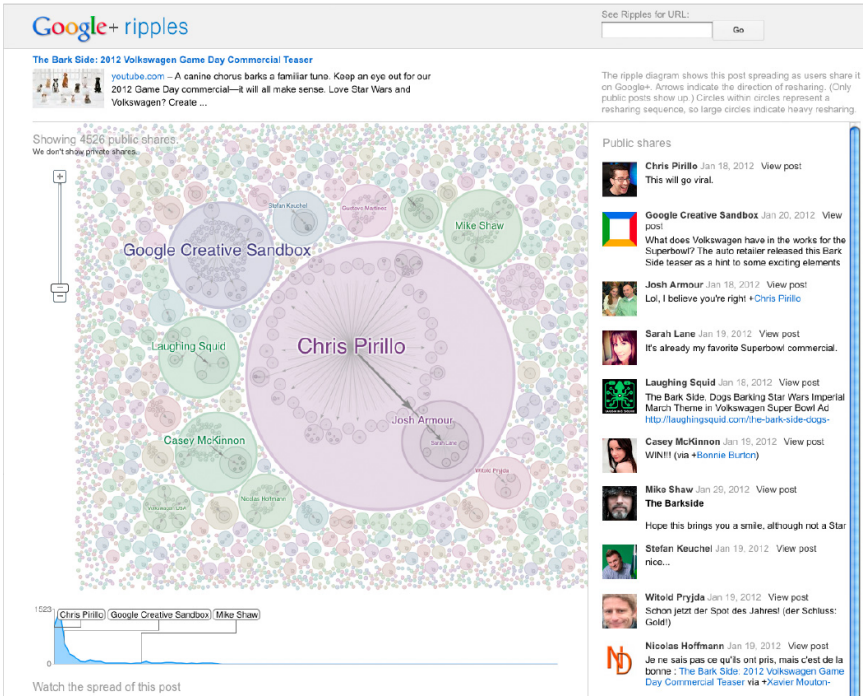


FIGURE 2.20: Information spreading in a social network visualized by Google+ Ripples..

remaining social graph of the social network. Google+ Ripples (Figure 2.20) thereby allows users to follow the spread of information for a post, from the original poster to the rest of the network. Whereas Google+ Ripples was published by the owner of the social network, which has access to the whole social graph, this is not the case for third-party app developers or for researchers, which typically only have access to the user's direct friends, and which cannot track the information flow thereafter. Nevertheless, it is possible to approximate the neighborhood of the direct friends using a graph generation algorithm [15].

The idea of raising privacy awareness by visualizing data flow has also been tested and evaluated in other domains like mobile smartphone apps [23]. Since the publishing of the most recent versions of the Android operating systems, apps do not require a special permission to access the internet, and can therefore send and receive data online as they want. It is therefore also hard for a user to see if and when the app is sending personal data like the location or phone ID to an advertiser. The app "Privacy Leaks" uses TaintDroid to detect whether this happens, and offers the user an overview screen for each app, denoting how often the app leaked personal

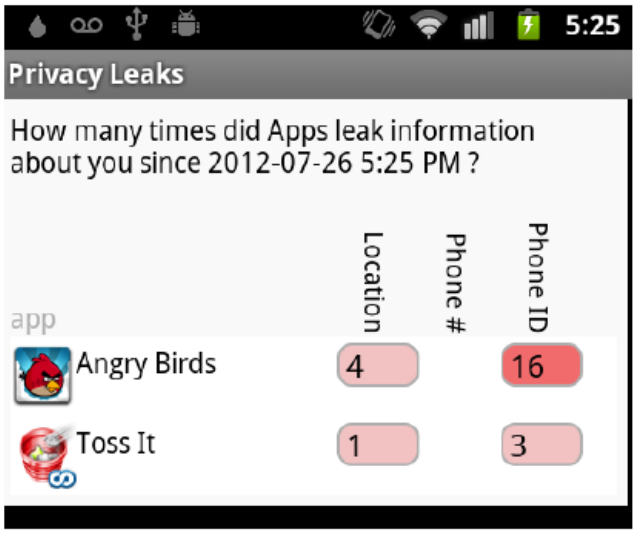


FIGURE 2.21: Screen of the PrivacyLeaks app, showing the amount of location requests for different apps.

information, and which information it leaked (Figure 2.21). On a similar screen, the user can also see to which advertising company the data was leaked and the number of times this happened. Although the researchers hypothesized that Privacy Leaks will lead the users to choose apps that leak less private information, the study did not show any significant effect of the software, although the usability and user acceptance was very high [23]. The main factor of influence is still the rating of the app, as well as recommendations of friends [23].

2.5.4 Grouping user interfaces

As stated in Section 2.3.2, grouping social network friends is a needed preparatory action in order to do narrowcasting, e.g. to disclose new data items like a social network post only to the desired audience. Grouping tasks (also called *categorization* tasks) in general have already been broadly investigated in research. There are several abstraction levels that can be used to implement a sorting task, each with its own advantages and drawbacks [280]. The most concrete and least abstract form of sorting is using the actual objects to be sorted for the task, called *object sorting*. Object sorting provides the highest level of detail of information about the object; on the other hand, it may also include details which are irrelevant for the sorting task and can thereby hinder or confuse the user. An approach that is a bit more abstract is *picture sorting*, where the objects are represented by corresponding pictures. Although this reduces the sensory input on one hand, it is possible to crop out unimportant details by this technique. Lastly, the most abstract form is card sorting, where the objects are represented by their name. Card sorting allows a precise limit on the information given to the user and thereby eliminates unneccessary details, but requires the user to recognize the object and being able to imagine it.

Apart from the abstraction level, the metaphor and degrees of freedom used in the UI also make it possible to differentiate among grouping user interfaces. Some UIs offer a predefined set of groups to be used (*closed* sorting), whereas others allow users to create new groups (*open* sorting). Whereas the former allows more comparable results, the latter is required if the data is highly individual and not comparable between subjects, and thus needs a higher degree of freedom for the user. Typical

FIGURE 2.22: Different card sorting metaphors [61]: Stacked card sorting (left) and explorer-based sorting (right).

metaphors used are the *stacked card sorting metaphor*, where the cards of the same group are stacked on top of each other (Figure 2.22 left), or *explorer-like* approaches (Figure 2.22 right). Which metaphor works best depends on the user group [61]: Whereas researchers like the explorer-based sorting best, end-users instead prefer the stacked card metaphor. Sorting has also been proven to work efficiently in VR environments [159]: In a study where users were given products to be sorted arriving on a conveyor belt (see Figure 2.23), they were generally able to discriminate and sort the products correctly if the visual difference between the products was large enough.

Kelley et al. found that users follow one of two strategies to group their friends [184]. Their study included three new user interface prototypes to perform the sorting task (Figure 2.24): an image sorting approach, where the users were given pictures of their friends to sort; a grid-based approach where the users were aligned on a grid with the possibility to mark them according to the desired friend group association with pens of different colors; and a folder-based approach where the users had to sort the images into folders using a UI similar to the Windows Explorer. As a fourth interface, the Facebook friend grouping interface was used as a reference.

The first of the two strategies Kelley et al. discovered in their study was the "by friend" strategy, where the friends are traversed one by one and each time a friend has to be assigned to a friend group which has not yet been created, the group is created on the fly. In contrast, in the "by group" strategy, the friend groups are created first and then populated by their members one by one. Furthermore, they stated that friend grouping should always be seen as a secondary task; UIs should therefore give users small friend grouping tasks from time to time rather than prompting the user to sort *all* friends at once. Friend groupings change over time; therefore, the friend grouping UIs should also try to get feedback on the correctness of the friend grouping from time to time. However, research on how the friend grouping task in social networks can be designed to be more enjoyable, apart from trying to increase its usability, has not been conducted so far.

FIGURE 2.23: Sorting of products arriving on a conveyor belt using VR [159].

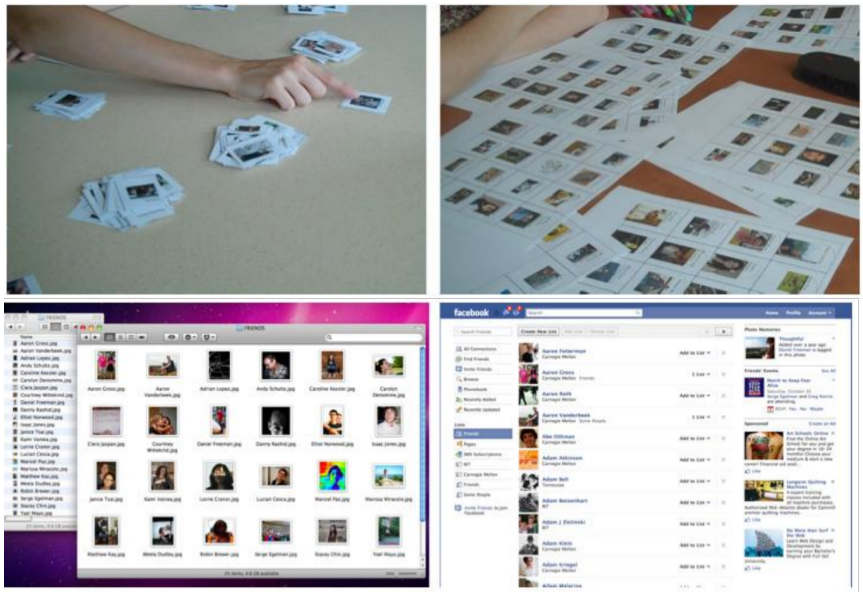


FIGURE 2.24: UI prototypes used by Kelley et al. [184] (from upper left to lower right): image sorting UI, grid-based UI, explorer-based approach, Facebook reference interface.

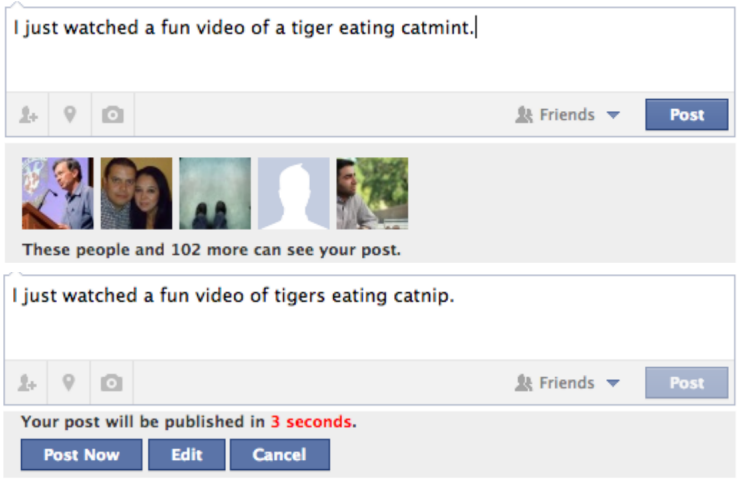


FIGURE 2.25: Audience + timer nudge, showing the user possible recipients of the post and the possibility to edit or withdraw the submission.

2.5.5 Consequence-based privacy user interfaces

Finally, a last form of graphical user interfaces helping users to improve their privacy is based on displaying the consequences of their privacy choice, enabling them to reflect on and possibly to change their privacy decision, also known as *privacy nudges*. Such a system has been proposed for Facebook posts by Wang et al., and is shown in Figure 2.25 [333]. The topmost text field is the current standard on Facebook, where the user can type in a new status message and define the friends which should be able to see the status update (in this case *all friends*). When the user clicks on "post", the status update is usually directly published. In the approach by Wang et al., this is not immediately the case: The UI first displays a portion of the recipients of the post using their profile pictures, and offers the user a three-second countdown to decide whether they still want to publish the post that way, or whether they want to change either the content or the audience that is able to see the post. An evaluation has shown that this change in the user interface indeed influences the privacy decision: A significant amount of users chose not to publish the post, or made some modifications to either the post content or disclosement settings, after a part of the audience, i.e. the possible consequence of their disclosement choice, was shown to the user [333, 334].

Privacy consequences can also be visualized using the user's smartphone home screen [286], as shown in Figure 2.26. Whenever somebody accesses the private data of the user, a new pair of eyes is drawn on the home screen, using the size of the pair of eyes to denote the frequency of the data request normalized by the closeness of the person, so that if a close friend makes data requests, the pair of eyes is grows more slowly than when a stranger requests the data, for example. The authors compared their ambient view with a detailed privacy page that showed the requesting persons and the number of requests, together with the possibility to restrict access for those persons. The results showed that using a detailed page requires more of the user's time, thus making it less suitable for checking one's privacy status on the go with the smartphone. The privacy settings created by the users of the detailed page were on average better than those of the users of the ambient notification, although the difference was not significant. However, comparing the privacy settings before

FIGURE 2.26: Visualization of the frequency of data requests on the user's home screen, either allowed or blocked

and after the use of the ambient notifications reveals that the disclosure policy was significantly improved through the usage of the notification application.

The same approach can also improve privacy for smartphone apps according to research [9]. Simply displaying excessive access requests to certain sensitive data items as shown in Figure 2.27, like location requests, combined with a permission manager that allows the user to adapt the permission settings for the apps mentioned in the UI, led 95% of the participants of the study to review their privacy settings, and 58% of them to further restrict the chosen privacy settings after receiving the feedback [9].

Also when sharing a location, using privacy nudges and thereby motivating users to adapt their privacy settings can help, according to Tsai et al. [326]. Their software "Locyoution" was able to display the user's own location as well as the locations of the user's friends, and included a history page, displaying recent location requests together with the possibility to restrict access accordingly. In a study comparing the amount of granted location requests and levels of comfort with being located, the researchers found that the amount of allowed requests was reduced from 44.9% before to 40.3% after the usage of the tool, although the difference was not statistically significant. However, the self-reported level of privacy concern of the user, as well as the comfort level of being located by friends and strangers could be significantly reduced by Locyoution [326].

As we have seen, there are several approaches for helping users improve their

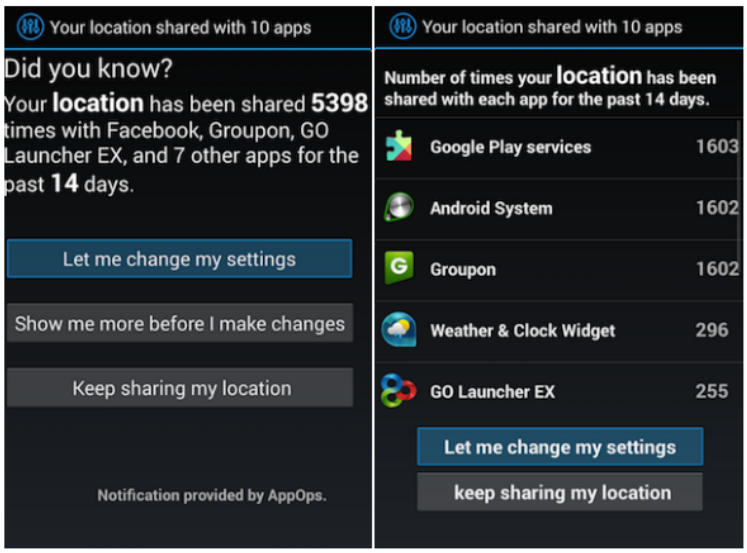


FIGURE 2.27: Privacy nudges for mobile app permissions.

privacy using UIs: Reducing the complexity of the settings and the UI by condensing and displaying the information to highlight the most important aspects; raising privacy awareness by highlighting the flow of sensitive information; supporting the decision process beforehand; or using privacy nudges, to nudge the user to validate and improve the privacy settings. In this thesis we will use ideas from all of these approaches: although not in a user interface, we will reduce the complexity of the privacy settings by suggesting personalized privacy settings using machine learning, that have only to be reviewed by the user. We will provide a user interface that allows the user to have an overview of their privacy situation at one glance using a radar metaphor in the example of the intelligent retail domain; and we will show how radar interfaces can help in selecting the right audience for a large amount of social network friends.

2.6 Machine learning and deep learning

Machine learning is a combination of statistical methods and algorithms that have the goal to perform tasks without explicit instructions by a programmer [37], and was first used by Arthur Samuel in 1959 [189]. It relies on recognizing patterns in the data or inferring missing data from a given data set. One general idea is to build mathematical models on labeled sample data (called "training data"), so they can later be used to deduct labels for unlabeled data. Other machine learning methods are specialized on working without training data, by using the inherent clusters inside the training data, for example for grouping data items together or for reducing the complexity of the data sets by removing duplicate data. Since 1959, many different machine learning methods have emerged, which can be categorized either by their learning style, or by their similarity in the way they are trained and used.

There exist three different types of learning methods [37]:

- **Supervised learning**, methods that require the algorithm is first given a training set including *labeled* data. The algorithm is then *fitted* to the given data, which means it learns from the training data: Whenever a new data item from the training set is processed by the algorithm, the predicted label and the correct label are compared. If both labels differ too much, the algorithm adapts its parameters so that the new data item as well as the already processed data items can be predicted with the smallest possible error. How this adaption process works, which parameters are adapted and how the difference between the labels is computed, depends on the actual algorithm type and will be discussed later. After the training phase, the fitted algorithm can be used to predict labels for unlabeled data items. Examples of supervised learning approaches are regression and classification approaches.
- **Unsupervised learning** methods, on the other hand, do not need a training set, and work on data which does not have a known result. They instead use the structure inherent in the given data set to perform different tasks, for example to cluster the data set into groups of similar data items (*clustering algorithms*), to remove redundant data items (*complexity reduction*) or to learn general association rules (*association rule machine learning*).
- **Semi-supervised learning** approaches are a mixture of both aforementioned learning styles. There exists a training set of some data items with a known result, but also some data items for which the result is not known. Similar to supervised algorithms, semi-supervised algorithms have the goal to make a prediction on unlabeled data, but in this case, the model has to first learn the structures present in the data to train the model and do the prediction on unlabeled data later. Some regression and classifications algorithms fall into this category.

For the tasks discussed in this thesis, we try to infer privacy settings based on users' personality and privacy measures. In all cases, we have a training set that has been recorded through a user study that can be used for fitting the algorithm; therefore the domain of *supervised learning* approaches is of interest in the scope of this thesis, and will be discussed in this section. In the following, we will describe some of the machine learning techniques that have been used later in this thesis, and how the algorithm selection is conducted.

2.6.1 Support vector machines

The original approach for support vector machines was proposed by Vladimir Vapnik and Alexey Chervonenkis in 1963 [166] and worked, in its original form, as a linear classifier. About 30 years later in 1992, a new method called the "kernel trick" was proposed that also allows support vector machines to be used for classifying non-linear patterns [46]. The approach of a support vector machine can be described best in a two-dimensional space, as depicted in Figure 2.28. Informally stated, a classification algorithm tries to separate the two classes of data items (blue and pink) by drawing a line between those two clusters. Mathematically, this imaginary separating line is a *kernel function* in support vector machines, the initial description of support vector machines used such *linear kernels* for the classification. The SVM algorithm then selects the line which has the largest margin between the closest points

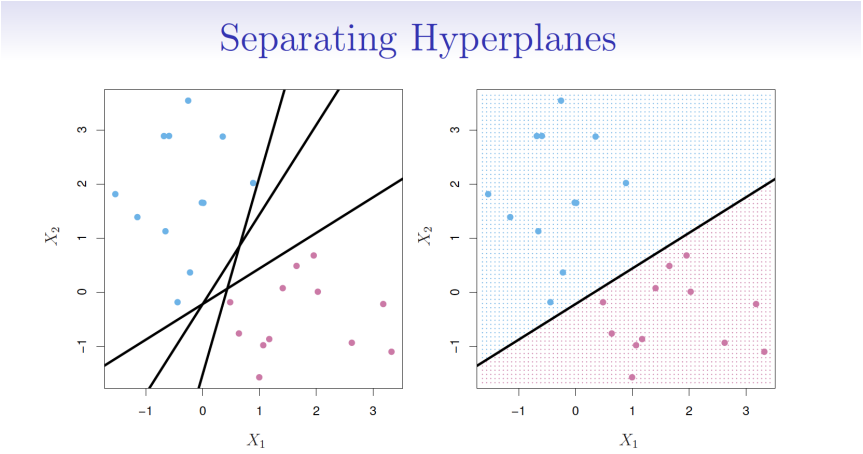


FIGURE 2.28: Different hyperplanes separating the data items in two-dimensional space. *Image source: An Introduction to Statistical Learning* [166]

of both clusters (see Figure 2.29), maximizing the generalizability of the classifier. This simple method is called the *maximum margin classifier* or *hard margin classifier*. Similarly, the same can be done using a plane in a three-dimensional space, or a hyperplane in an n-dimensional space.

However, it is not always possible to separate two clusters using a simple line, as shown in Figure 2.30. For this purpose, the authors of the original SVM approach proposed a method that also allows the misclassification of some of the training items, called a *soft margin classifier* or *support vector classifier* [75]. This type of support vector machine introduces a regularization parameter *C* that denotes the amount of allowed misclassifications. However, choosing the correct C is not always trivial, as a small C (only a few misclassifications allowed) will on one hand reduce the bias, but on the other hand may introduce a high variance. On the other hand, a high C leads to a higher generalizability, but also increases the bias. Therefore, the authors later introduced an approach called the "kernel trick" to solve the problem of separability in an n-dimensional space, by introducing polynomial kernels (see Figure 2.31) and by transferring the problem space into a higher dimension, where the separation is again possible using linear or polynomial kernels (see Figure 2.32).

Support vector machines are especially suitable if the clusters can be separated well, and can also handle non-linear data using the kernel trick. On the other hand, it is hard to determine the cause of misclassification problems, as the use of kernels for separating the data makes it hard to interpret the algorithm parameters that are estimated during the learning phase.

Apart from classification problems, support vector machines can also be used for predicting label sets with an unlimited amount of labels, which corresponds to a regression algorithm called *support vector regression* (SVR) [93]. Support vector regression uses the same principle as support vector classification, but this time the distance from the data points to the hyperplane is minimized. Also, SVR algorithms can use either a linear or a polynomial kernel for the approximation.

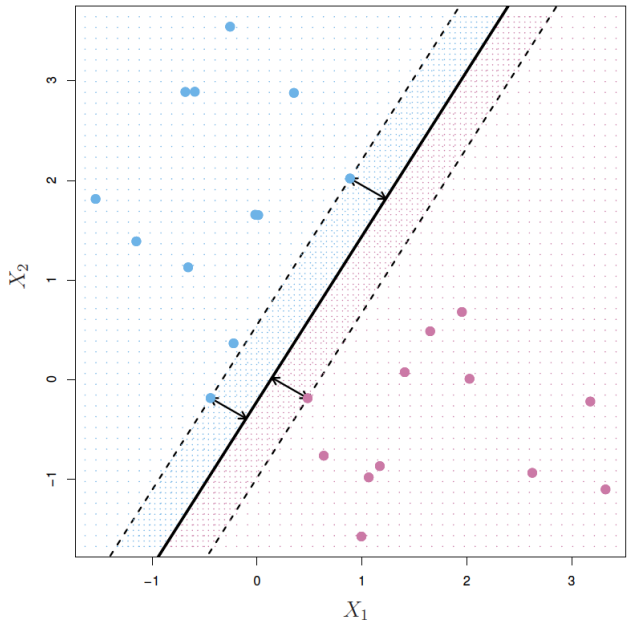


FIGURE 2.29: Margin between separating line and clusters that has to be maximized by the algorithm. *Image source: An Introduction to Statistical Learning* [166]

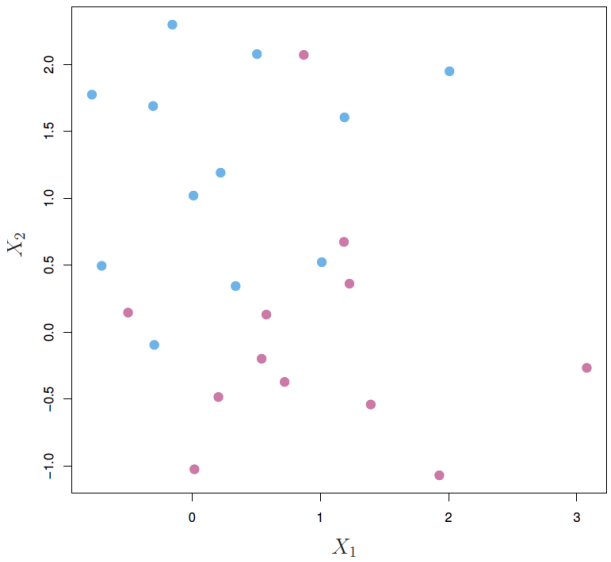


FIGURE 2.30: Clusters that cannot be separated by a maximum margin classifier. *Image source: An Introduction to Statistical Learning* [166]

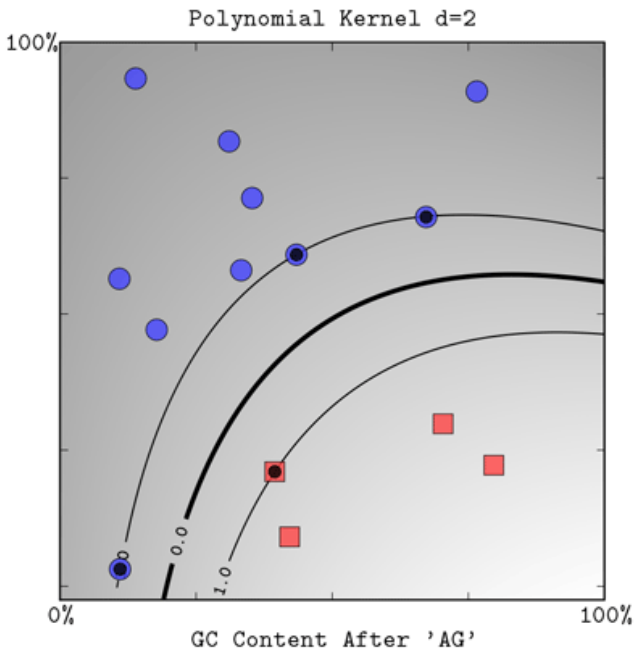


FIGURE 2.31: Polynomial kernel separating two clusters. *Image source: An Introduction to Statistical Learning* [166]

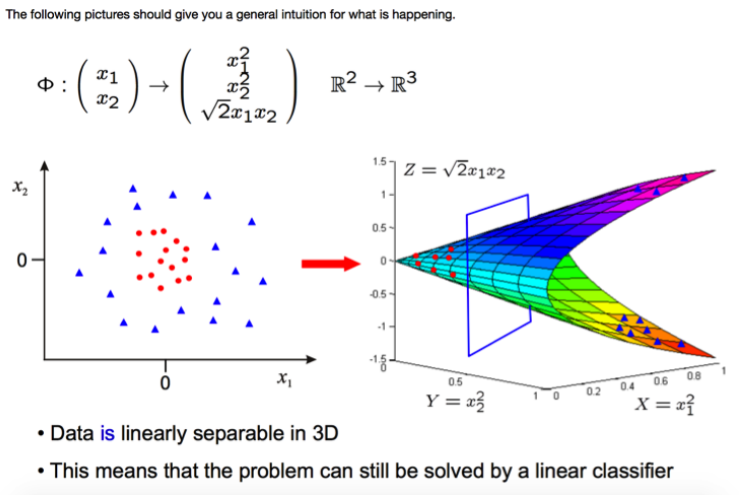


FIGURE 2.32: Solving a clustering problem by transferring the data into a higher dimension. *Image source: http://svmcompbio.tuebingen.mpg.de/img/poly.png*

2.6.2 Ridge regression

A problem that is often experienced using simple linear regressors such as the support vector regressors is the problem of *multicollinearity* [319]. Multicollinearity exists when two or more of the input variables in the data set have a strong correlation with each other. Let's say X and Y are input variables with a strong correlation. That means there is no data set in which X changes while Y stays more or less constant; X and Y are always changing together, making it hard to precisely estimate coefficients for both variables, as they cannot be observed isolated from each other. Even small changes in the data set can lead to a large change of the estimated regression coefficients, making the regression model highly unstable, and leading to a reduced precision compared to models without any correlation in the input. Overfitting is often caused by redundant input parameters that are correlated to each other. Furthermore, the calculated significances for the different input variables are not reliable; the more variables that are correlated with each other, the more the t-values and the significance of the input variables is reduced. That means the statistical analysis of the model can only state how good the overall accuracy of the model is, but it cannot reliably state how well the individual parameters of the prediction perform.

In practice, multicollinearity can be detected, in its simplest form, by observing the regression coefficients of the other input parameters when a new feature is added to the model. In the ideal case, the other coefficients should not change; if some of them change significantly, there might exist a multicollinearity with those parameters. In order to be able to also include input variables that are multicorrelated, an approach has been invented called *Tikhonov regularization* or *ridge regression* [320], named after its inventor Andrey Tikhonov. Broadly speaking, ridge regression allows a better estimation of the coefficients of multicollinear model parameters at the cost of an increased, but tolerable bias. Let's say that A is our input matrix, where each row contains one set of observed values and b is the vector containing in each row the corresponding value we want to predict. Then a regression has to find an x so that $A * x = b$. A simple regression would then try to minimize the distance of the regression hyperplane towards the data points, e.g. $\|A * x - b\|$, where $\|\cdot\|$ is the Euclidean distance. The ridge regression instead adds a regularization term to the equation, so it optimizes $\|A * x - b\| + \|\Gamma x\|$, where Γ is called the Tikhonov matrix. Usually, Γ is chosen as a multiple of the identity matrix ($\Gamma = \alpha * I$), so that solutions with smaller norms are preferred, known as L_2 *regularization*. By that means, it is possible to shrink the size of the coefficients based on the square of their magnitude, reducing the problem of overfitting. The amount of shrinking of the coefficients is controlled by a regression parameter λ. If $\lambda = 0$, the ridge regression is equal to a least squares regression, whereas all coefficients are shrunk to zero if $\lambda = \infty$. Usually λ is chosen between 0 and 1, typically starting with 0.01 as a starting value.

Ridge regression is especially suitable for input data where the probability of correlation between the input parameters is high. The predictions that are performed later *all* depend on individual measures like personality or privacy desire; sometimes there are multiple, different measures included from the same topic, for example multiple privacy measures. Therefore the possibility of experiencing multicollinearity in the input data is high in the problems discussed throughout this thesis. However, we apply heuristics to eliminate redundant input data; therefore we can neither state that our data sets are free from multicollinearity, nor can we say that multicollinearity is always present. We will therefore check for every single domain whether a regularized method like the ridge regression is needed or whether unregulated methods like support vector machines are sufficient, as we will see later

in Chapters 4, 5 and 6.

2.6.3 Regression with categorical input parameters

The regression algorithms presented up to this point all have one thing in common: Input parameters always have to be continuous scales or ordinal variables with a more or less equal distance between their values. However, regression problems often also depend on ordinal variables where equidistance cannot be guaranteed, or even on categorical variables, which cannot, without further processing, be used as a regression input [37]. In this case, if the categorical variables have to be included in the set of regression parameters, the data scientist has to first recode the variables, for example into a binary representation. For this purpose, the number n of categories of the categorical variable V is counted. Then, for each possible category $C_1, ..., C_n = C$, one binary variable $B_1, ..., B_n = B$ with a value of either 0 or 1 is created. The value of each binary variable is calculated as follows:

$$\forall b_i \in B, c_i \in C : B_i = \begin{cases} 1, if V = c_i \\ 0, if V \neq c_i \end{cases}$$

That way, categorical variables can be transformed into a set of binary variables that can be used as input parameters for the regression. However, recoding variables using this approach, also known as *one-hot encoding* [148], leads to the effect that the different outcomes of the same categorical variable are coded into different binary variables for the regression; therefore each outcome is observed as an individual parameter (or separate case) by the regression algorithm, neglecting the connection between the binary variables. Therefore a recoding in binary variables can lead to a reduced expressiveness of the categorical variable. Another possibility is to recode categorical variables into a single variable, for example by using *ordinal encoding* [148]. In this approach, each category is assigned an integer value depending on the position of occurrence within the data set. The first category is assigned 1, the second 2, and so forth until the nth category is replaced by the integer n. However, the values assigned to the different categories are arbitrary; nevertheless, the regression algorithm expects the integer values to be in a logical order, leading to a misinterpretation by the regression algorithm and a loss of precision connected to this. Furthermore, as the encoding depends on the order of categories in the data set, the categories will receive different encodings when the data set is shuffled or when another data set is used, making it hard to compare the regression results between data sets.

To avoid this problem and to automate the conversion of categorical and ordinal variables into scales, a technique called categorical regression (CATREG) has come into focus lately [226]. CATREG transforms ordinal and categorical variables into scales by using optimization algorithms to find the optimal order for the categories and also the best distance between them. Also for ordinal measures, where the distance is not necessarily equal, CATREG can apply its optimization algorithm to find the optimal distance between each of the discrete levels, so that the prediction accuracy is maximized. For this purpose, CATREG utilizes nonlinear transformations of the input variables and different rating functions to determine the quality of the chosen transformation. Based on the statistical results and the standard error of the regression model, the algorithm adjusts the order of categories or the distance between categories, and performs another test run. This procedure is repeated until either the maximum number of iteration steps is reached, or the result improves by

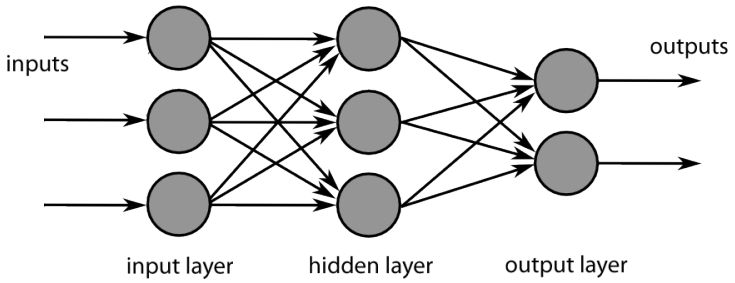


FIGURE 2.33: Layers of a deep learning algorithm *Image source: commons.wikimedia.org/wiki/File:MultiLayerNeuralNetwork_english.png*

less than a given threshold ϵ. Similar to a ridge regression, CATREG also offers to apply regularization methods on the feature coefficients to avoid overfitting caused by mutlicollinearity. As a standard, Tikhonov regularization is applied, but other methods like Lasso and elastic net are also available.

Within this thesis, it is often the case that ordinal, and sometimes also categorical, variables have to be used as input features within a machine learning prediction. Again, whether encoding methods like one-hot encoding lead to the best results, or whether specialized regression methods like CATREG should be applied, depends on the actual data set and will be discussed in the respective sections later.

2.6.4 Deep learning

The contents of this subchapter are based on the book "Deep Learning" by Ian Goodfellow [136]. The aforementioned algorithms are all very suitable to solve problems which are intellectually hard for humans to solve (for example because they require a high amount of complex calculations), but which can be described formally very well. However, there are other problems like face recognition or text recognition which can be solved intuitively by humans, but which are hard to describe formally and therefore hard to solve using conventional machine learning algorithms. For solving these problems, a different machine learning technique called *deep learning* [136] is usually applied. Briefly stated, deep learning divides the problem to be solved into a large number of small steps, which are realized by *layers* inside the deep learning model. As depicted in Figure 2.33, a deep learning algorithm consists of several types of layers. The input data (for example the pixels of an image) is fed into the input layer. After the input layer, several *hidden layers* process the data in a way that is determined by the deep learning algorithm, before the data reaches the output layer, which delivers the result of the analysis (for example the name of the person that was recognized in the input picture).

The tasks that are conducted by each of these layers are realized in *units* which are designed by the programmer. Similar to the aforementioned machine learning algorithms, deep learning approaches are also trained by sample data, for example photos annotated with the name of the person that is shown in the image. As we have seen in the previous sections, machine learning algorithms typically rely on input features. The quality of the input features, especially the presence of random variables and the variance of the data, significantly influences the precision of the prediction. Data preparation and isolation of the features that are important for the prediction (e.g. the independent variables) is therefore often needed prior to the training of the ML algorithm. For some cases where the independent variables are

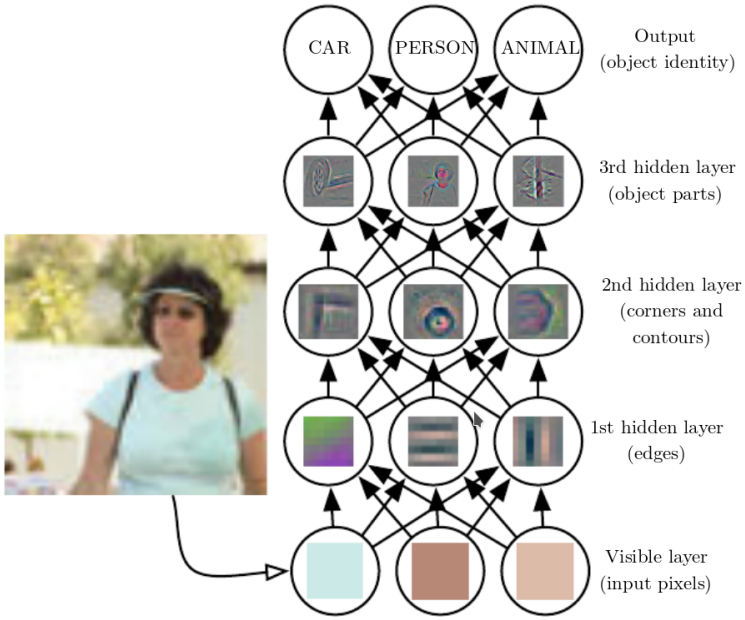


FIGURE 2.34: Example of an MLP for object classification in images.
Image source: [136]

known or where the amount of random variables is low, those approaches work very well. However, if the variance of the data or random variables is high, the algorithms deliver a low prediction precision. Let's say the algorithm has to recognize a person in a given image. Images contain a lot of random variables, for example the view angle, distance, lightning conditions, facial expression, and even the hairstyle or clothing of the person on that day.

Feedforward deep networks

The most essential example of a deep learning model is the feedforward deep network, also called the *multilayer perceptron* (MLP). An MLP can be seen as a sequence of mathematical functions (the hidden layers) that are applied subsequently on the input data. Figure 2.34 shows an example of a multilayer perceptron for object classification in images. The input layer is fed the pixels of the image. The next, first hidden layer detects edges in the image. Based on this data, the second hidden layer detects corners and contours. The third hidden layer uses the corners and contours for identifying object parts, whereas the final output layer delivers the textual representation of the recognized objects. As the example shows, each of the layers performs a relatively simple operation on its own. Given the pixels, it is easy to detect edges by comparing the brightness of neighboring pixels. Given the edges, it is easy to find the contours as a collection of edges. By finding specific combinations of corners and contours, it is easy to identify object parts. Finally, the combination of object parts allows it to easily identify objects. To conclude, by dividing the complex task of object identification into multiple easy parts, the MLP is able to perform the complex task.

In contrast to conventional machine learning algorithms, the increased complexity of deep learning methods, involving hundreds to thousands of processing layers, needs a significantly larger amount of training data in order to perform well. The roots of deep learning algorithms can be found in the 1950s. However, they

became popular only starting in the 1990s, when the amount of training data available was sufficient to train the algorithms correctly. In addition to the amount of data, the computational ressources needed for deep learning are also significantly higher compared to other ML approaches, which is another reason why they became popular only 40 years after their invention, when the computational hardware was fast enough to run them. The larger and more complex the deep learning model, the more precise the predictions can be, but also more computational ressources are needed.

Similar to the machine learning algorithms discussed earlier, deep learning algorithms also have a problem with generalization, meaning that they work well on the training data, but not on the test data. Therefore, similar to machine learning, there also exist *regularization* methods for deep learning that increase the precision on test data, possibly at the cost of prediction precision of the training data. Similar to a ridge regression, L^2 parameter regularization can also be applied to the coefficients of the deep learning functions in each unit of the deep learning model to avoid overfitting. Another approach that is used to avoid overfitting is to create fake data that is added to increase the size of the training data set, also known as *dataset augmentation*. Input data often has several known variations for which the deep learning model should be made robust, for example the rotation or lightning conditions of an image for face recognition. In that case, additional training images can be generated by rotating the images of the training set, or by changing the illumination. Injecting image noise is another option for generating additional images. Apart from generating additional training data, another solution for increasing the robustness to variance of the deep learning model is to introduce noise to the parameters of the unit functions. By that means, additional variance that is not present in the training data can be incorporated into the trained deep learning model. There are many other regularization methods like parameter tying, early stopping and multitask learning, and methods that combine a multitude of models together for the regularization (for example bagging and other ensemble methods), which will not be discussed here for the sake of brevity.

Convolutional neural networks (CNNs)

CNNs were first described by LeCun et al. as a method for the recognition of handwritten ZIP codes [198]. CNNs are a specialized kind of neural network that works on two- or three-dimensional, grid-like data; they are therefore especially suitable for applications like time series analyses and image processing. Convolutional networks are based on two different operations that are conducted one after another in *layers*. Those two types of layers are named *convolution* and *pooling* and are described within the next paragraph. Especially for classification tasks, the output of a convolutional network is often fed into a traditional MLP for the actual classification task on the preprocessed data.

Convolution layers [136, pp. 327] use a convolution function to process a region of input data using a convolution matrix (also called a *kernel*). A convolution operation S for a two-dimensional input I using a two-dimensional convolution kernel K is defined as follows:

$$S(i,j) = (I * K)(i,j) = \sum_m \sum_n (I(m,n))K(i-m, j-n)$$

That means for computing the output of the component (i,j), not only is the pixel (i,j) in the input used for the computation, but also its neighbours.

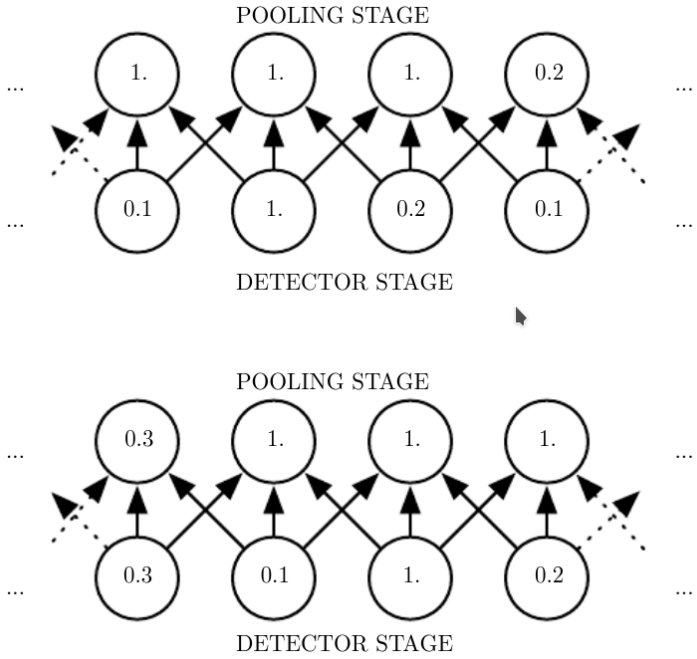


FIGURE 2.35: Pooling being invariant to small translations: Whereas the input is shifted one pixel (top image to bottom image) and thus causes all four items in the input to change, only two are affected in the output. *Image source: [136, p. 337]*

The most important difference from an MLP is that conventional neural networks like an MLP use a matrix for the computation of the output which has the same size as the input matrix. That means, if an image to be processed has $x * y = M$ pixels, the processing matrix would also have to have the size $N = x * y$. The processing time for computing the matrix product in a MLP would be therefore be in $\mathcal{O}(M * N)$. In contrast to that, convolutional networks use sparse matrices, meaning the matrix that is applied to each each pixel is significantly smaller compared to a conventional neural network and therefore faster to compute, reducing the computation time to $\mathcal{O}(k * M)$ where $k \ll N$. In addition to this, convolution uses parameter sharing, meaning not every data point is assigned its own separate wheight, but weights are shared among several data points, thereby further reducing the memory needed for the computation. The sharing of parameters also causes the convolutions to be *equivariant* to translations [136, pp. 330].

Convolutional networks usually execute several convolution steps in parallel at the beginning and feed them into a non-linear activation function like a rectified linear activation function (*detector stage*). After this, *pooling* is used to further modify the output [136, pp. 335]. Whereas the output of a convolution has the same size as the input, a pooling operation combines several nearby inputs into a single output, thereby reducing its size. Max pooling, for example, selects the largest number of the input region as the output. Other pooling functions return the (weighted) average of the rectangular neighborhood or the L^2 norm. Using pooling, the representation is made invariant to small translations, as the output of the pooling function does not change (see Figure 2.35). Which pooling function should be used depends on the data and the situation [47].

2.6.5 Choice of the right algorithm

Unfortunately, there is no single algorithm that works best on *all* supervised learning problems (this state of affairs is called the *no free lunch theorem*) [37]. The choice of the best algorithm depends on several issues that have to be traded off; in some cases even a simple trial-and-error procedure is the best way of finding the best algorithm for the given problem. The four major issues to be considered for selecting a supervised learning algorithm are the *bias-variance trade-off*, the *function complexity and amount of training data*, *dimensionality of the input space*, and the *noise in the output values*.

The *bias error* in a supervised algorithm is often caused by biased training data, e.g. training data that is not representative and that is missing a lot of important information [126]. If an algorithm has a high bias, it can perform extraordinarily well on data items that have a similar bias as the training data, but will perform poorly on data items that are not present in the training set. A high bias or high bias error is also denoted as *underfitting*, which means that the algorithm is too specialized on the given (biased) training data in order to work for a more general, unbiased data set. On the other hand, the *variance error* arises from the sensitivity of the algorithm to small fluctuations in the data. In this case also known as *overfitting*, the algorithm is modeling and predicting the random noise inside the training data, reducing the prediction accuracy of the model. The goal is usually to reduce both bias and variance error inside the trained model. Unfortunately, bias and variance contradict each other: A data set with a minimal bias has a higher variance of the algorithm parameters across different samples, and vice versa. A countermeasure for this so-called *bias-variance dilemma* is the usage of a *bias-variance decomposition*, which tries to automatically predict both bias and variance. Bias-variance decomposition was originally implemented for a classification method called least-squares regression, but regression methods like LASSO and ridge regression also have regularization methods that use the bias-variance decomposition to reduce the sum of both errors to a minimum.

The second important thing when choosing the ML algorithm is the choice of the complexity of the algorithm, which has to fit the complexity of the data [37]. If the data is not very complex, a simple algorithm, for example using a linear regression, can also provide good results on a small, biased data set. As an example, let's say we want to build a machine learning model that predicts the speed of a falling object based on the falling time of the object. As we know from our physics education, the speed can be calculated as $v = g * t$ where g is the constant gravitational acceleration, forming a simple linear equation. Similarly, if we are using a linear regression to solve the problem, we need only two rows of training data to get good results, as all other data points will follow the same linear regression curve. On the other hand, if we want to predict complex scenarios like life expectancies based on lifestyle habits, more complex machine learning algorithms should be preferred. Unfortunately, it is often the case that the complexity or the expected shape of the regression curve is not known beforehand; therefore, the machine learning algorithm with the optimal complexity can be found only by trial and error.

The third factor to be considered when approaching a machine learning problem is the dimensionality of the data [37]. Independent from the actual machine learning algorithm, a precise prediction gets harder the more input features the algorithm is given within the training set, even if the outcome depends only on a small number of the provided input features. The higher the number of features (the *dimensionality of the input space*), the more the algorithm can be distracted by unnecessary input

features that lead to an increased variance in the predicted outcome, e.g. to an *overfitting* of the algorithm. Because of this reason, when the input space is large, one should try to reduce the number of input features to a minimum, for example by using *dimensionality reduction*, or by using heuristics like the *wrapper subset selection* (WSS) to iteratively find good input features while discarding irrelevant ones.

As a fourth factor, it is important to keep an eye on the noise of the output values in the training set. Especially if the output data is human input data, for example from a user study, the data scientist has to first check the output values for plausibility and noise, and remove outliers before applying the machine learning algorithm. If the output data is too noisy and not sanity-checked beforehand, the trained model is prone to overfitting, as it is trying to also model the noisiness of the data from the training set.

Lastly, there are also other factors that have a minor, but still significant influence on the choice of the algorithm and the precision of the prediction. Based on the heterogeneity of the data, e.g. whether all input features are scales, or whether ordinal or even categorical features are also present, the data scientist has to decide whether she will use simple algorithms like vector machines or linear regression that can use only scales as an input, or whether more complex approaches like categorical regression (CATREG) should be employed to cope with ordinal and especially categorical variables. Some methods like linear regression, logistic regression, or distance-based methods are prone to redundancy in the data, which should therefore be reduced to a minimum.

Within our studies, we always manually checked the study data in order to remove outliers, and used control questions to automatically remove participants that did not fill out the questionnaire earnestly before using the study results as training data. The algorithm we chose depended often on the type of input data that we used; we therefore applied simple algorithms like support vector machines, as well as algorithms reducing the bias-variance error like ridge regression, up to a categorical regression that we used for the location sharing domain, where several categorical and ordinal variables are present. We used wrapper subset selection heuristics to reduce the dimensionality of our data to a minimum. Details on the methods that were applied on the different domains will be given later, in their respective chapters.

Although deep learning models have been shown to be very precise, they require a high amount of computational power and a large amount of training data. As the training data used within this thesis had to be collected by user studies, the amount of data was sufficient only for traditional machine learning algorithms. Therefore we see the investigation of using deep learning for recommending privacy settings based on user personality as a task for future research.

2.6.6 Evaluation of a trained model and cross-validation

Often only one data set is available for training and evaluating a machine learning model. Therefore, to evaluate the precision of the model, the data is separated into two parts. The first part is called the *training set* and is used for training the model and for adjusting the parameters. In supervised learning, the machine learning algorithm is given both input features, as well as the values of the dependent variables the algorithm has to predict. Typically, 90% of the data are assigned to the training set [125]. After the model is trained, the remaining 10% of the data, called the *validation set* or *test set*, is used for validating the trained model. For this purpose, the model is given only the input values of the validation set, and has to predict the values of the dependent variables. To measure the quality of the algorithm, the

averaged difference between the predicted and the actual value is computed, called the *standard error*.

However, if this procedure is done only once, it is still possible with a small standard error that the algorithm would perform worse with another data set, i.e. the generalizability of the trained model cannot be assured with this approach. For this reason, a technique called cross-validation is typically used when evaluating a machine learning model. There are two families of cross-validation methods: the exhaustive cross-validation and the non-exhaustive cross-validation methods [125, 187]. Whereas exhaustive methods try out *all* possible partitions of the data into training and test data, non-exhaustive methods do *not* test all possible combinations and can instead be seen as a form of sample testing. In most cases, the number of data items and hence the the number of possible partitions is too large to allow exhaustive cross-validation within a reasonable computing time [125].

There are two exhaustive cross-validation methods, namely *leave-p-out cross-validation*, and a specialized form of leave-p-out cross-validation with $p = 1$, called *leave-one-out cross-validation* [59]. In these methods, p rows of the data set (or *data items*) are used for the test set, while the remaining data items remain in the training set. This is done for *all* possible combinations of dividing the data set into p items for the test set, and the remaining items for the training set. Similarly for the leave-one-out cross-validation, one data item is always kept for the test set, while the others are used as a training set.

The most frequently used non-exhaustive cross-validation method is the *k-fold cross-validation* [50], especially in its specialized form with $k = 10$, called *ten-fold cross-validation* [274]. In this approach, the data is partitioned into k folds of an approximately equal size. One of the folds is always the test set, while the others are used for the training set. The procedure is repeated for each of the k folds, and the results are averaged. If $k = n$, i.e. if the amount of folds is equal to the amount of data entries in the data set, this approach is equal to leave-one-out cross-validation [151]. In contrast to other methods that use random sampling, k-fold cross-validation makes sure that each data item is part of the test set exactly once.

Apart from the more controlled k-fold cross-validation approach, another non-exhaustive cross-validation method that is often used is repeated random sub-sampling validation, also known as Monte Carlo cross-validation [94]. In this method, all data items are randomly assigned to the training or the test set [194]. The size of the training and the test set is defined beforehand. The procedure of dividing the data set into training and test sets, training and evaluating the machine learning model is then repeated n times, and the results are averaged over all runs. In practice, the procedure is repeated until the change of the averaged result is below a certain threshold. As the number of runs approaches infinity, the results of the repeated random sub-sampling validation converges to the results of a leave-p-out cross validation [94]. In contrast to k-fold validation, the size of the training and test set and the number of runs can be freely chosen. On the other hand, it may occur (and will occur with increased n), that the test sets and training sets of different runs will overlap, or that some of the data items will never be assigned the to test set. In this thesis, both ten-fold cross-validation as Monte-Carlo cross-validation will be used.

2.7 User modeling and cross-domain user modeling

As mentioned in earlier sections of this chapter, some of the privacy recommender systems offer privacy settings based on a user stereotype. In order to find the correct stereotype, these approaches first have to build a *user model*. In this section, we will first discuss how user modeling is performed, and how user models can be transferred to other domains (*cross-domain user modeling*), which helps us to predict privacy settings for a domain based on the privacy settings from other domains, as discussed in chapter 7.

2.7.1 User modeling

The major part of the content of this section is based on the chapter "User modeling" from the "Handbook of Human Factors in Web Design" [173]. The main goal of user modeling is to adapt the system to the individual user's needs, meaning the system needs to "say the 'right' thing at the 'right' time in the 'right' way" [117]. This means that the system first has to gather data from the performance or behavior of the individual user to either predict user behavior in a human-machine system (*predictive models*), or to adapt the user interface to the individual user's needs (*personalized models*) [173, pp. 3]. Which data is included in the user model heavily depends on the domain and the task for which the user model is developed.

Predictive models

The simplest form of predictive models is static user models. These models are typically gathered by performing a task analysis, for example using the GOMS technique [56]. For the GOMS technique, *goals* and *subgoals* of the task are analyzed, as well as the *methods* that are used to achieve these goals, and the *operators* that are carrying out the task. However, those static predictive models involve no simulation and therefore also cannot predict the time needed for conducting the task. Once modeled, they cannot adapt to the current situation or changes in general. Dynamic predictive models, in contrast, also model the cognitive and motor performance of the user which has been observed in the task analysis, for example, allowing them to mimic more realistic user behavior by dynamically adapting the user model [173, pp. 8]. More advanced user model architectures also incorporate learning skills, and are thus able to automatically learn from user decisions which are not explicitly modeled by the developer and thereby also able to infer such user decisions in the future, for example using machine learning techniques [234]. Apart from being static or dynamic, some of the predictive models use *stereotypes* of users, where groups of similar users are clustered together and represented by the same user profile in the user model, whereas other predictive models are *highly adaptive*, meaning that every user is assigned her own user profile in the user model, depending on her individual performance [158].

Personalized models

Personalized user models are an important part of adaptive user interfaces [173, pp. 16]. In contrast to adaptable user interfaces, which allow the user to customize the UI to her needs, for example by changing the keyboard layout or spell checking language, adaptive UIs try to automatically infer when user preferences change, and perform the adaptation automatically, based mostly on a personalized user model. Some of the approaches use direct input from the user for creating user profiles, for

example using questionnaires [116]. Based on the answers, the user can be matched to other users in the database, allowing the UI to be adapted according to their preferences. This technique, called *collaborative filtering*, assumes that users with similar characteristics will also act similarly and prefer similar UI designs. Apart from collaborative filtering, the adaptation can also be performed based on observed usage characteristics like mouse or eye movements, or keyboard usage, for example, in order to recognize that the user is experiencing difficulties and needs help, which can then be incorporated into the UI. Also, *how* the UI elements are accessed can be used to infer which task the user is trying to perform. A museum website can, for example, be used for gathering information about specific exhibitions that are of general interest for the user, to provide a virutal tour of the museum, or to help the user find interesting details about exhibits he found interesting during his visit. Based on the detected purpose of the visit to the website, a different presentation of the information may be suitable [173, pp. 22]. Another application of personalized user modeling is to support users in their tasks, for example by ranking search items higher when they might be of more interest for the user, or to limit the number of displayed choices in a UI based on the user profile.

User models are used for a wide variety of applications, for example tutoring systems that adapt to user abilities and learning skills [173, pp. 24], to facilitate interaction amoing users or to adapt the display of electronic books to the user's needs [173, pp. 28], or to implement agents that act autonomously on the user's behalf [173, pp. 30]. Personality stereotypes have also been used for recommending privacy settings in research [128]. However, using highly adaptive user models to offer each user their own privacy recommendations, has not been part of research so far. The privacy recommenders discussed throughout this thesis all rely on highly adaptive user models for their recommendations.

2.7.2 Cross-domain user modeling

Recommender systems described in earlier sections of this chapter often rely on input data that they require from the user, for example her personality measures, privacy measures, or user interaction data like privacy settings chosen in the past. However, if a new user enters the system for whom no data is available so far, for example because he has just registered on a social media website, those approaches fail. Only over time, when the user generates more and more data to be used by the recommender system, can the recommendations start to become precise. This problem is denoted as the "cold start problem" in recommender systems [285]. A similar problem arises when domain data is present for the user, but there is not enough for a single-domain recommender system to be able to make a good prediction, called the "sparsity problem" [243, 204].

Both problems are not new and have been considered in research often. Usually the idea is to use user models from other domains as an input for the prediction, for example by using privacy settings on other similar social media platforms to do the prediction (for example using privacy settings from Google+ to predict privacy settings on Facebook), or by using input from other domains, for example using the choice of app permission settings to derive privacy settings for a social network platform. This so-called *user model transfer* has led to mixed results [348]. On one hand, the precision is better using single-domain recommender systems; on the other hand, cross-domain recommender systems have the ability to predict in more than one domain, which can lead to an increased engagement and satisfaction

of the user [5]. Therefore, user model transfer should be used when a single-domain recommender system has insufficient data about the target domain [282].

Collaborative filtering

Cross-domain recommender systems can be divided into two groups of recommenders: those which are based on *collaborative filtering*, and *content-based* approaches. Collaborative filtering approaches [33, 55] rely on the assumption that if a user A behaves similarly to another user B in one domain, then she will behave similarly in another domain as well. The idea of cross-domain user modeling using collaborative filtering is therefore if data for user A is only available only in a domain D_1 and a recommendation has to be done for domain D_2, to find a similar user B for whom data from both D_1 and D_2 is available, and who behaves similarly to user A in D_1. A is then given recommendations according to B's behavior in D_2. In the case of privacy settings, this means that the recommender system searches for a user B that matches A's privacy settings in D_1, let's say in the social media domain, and uses B's privacy settings in D_2, let's say her mobile app permission settings, to recommend similar permission settings to A. Technically, these approaches are mostly based on machine learning approaches like nearest-neighbor. Let's say, for example, that the data items to be matched are ratings for movies in different domains. Then, the collaborative filtering algorithm treats the rating for the movies in one domain as one vector, and uses a nearest-neighbor algorithm to find a user that gave similar ratings to those movies. Then, the algorithm predicts ratings for movies in other domains (or books from the same or another genre, to mention another example) by averaging the ratings of other users who are similar according to the nearest-neighbor algorithm. The user is then given recommendations based on these predicted ratings.

Content-based approaches

In contrast to that, *content-based* approaches do not rely on numerical data, like ratings for example, but instead use the actual content for matching similar items, for example keywords, social tags or semantic properties. More specifically, the content of each item is denoted by a set of features $F = F_1, ..., F_n$. For each of these features, an item is assigned a vector consisting of real numbers, where the i-th component of the vector denotes the *weight*, i.e. the relevance of the corresponding feature F_i for the item. Similarly, each user is given a vector containing real numbers that describe how much he likes or dislikes this feature. An overlap between two domains that can be used for cross-domain prediction exists if some of the features are present in both domains A and B, i.e. $F_A \cap F_B \neq \emptyset$. If so, the recommender system uses the features from F_A that are known from the user, and uses the matching features from F_B for the recommendation. Instead of using keywords for the matching, some of the content-based approaches are based on *social tags* that are given by the users, so instead of using a feature vector with a predefined set of features, the set of features consists of a vocabulary T, which contains the social tags used in the environment (for example a social media page). In those approaches, the user is characterized by the tags she gives to the items (e.g. videos, post entries, etc.) she publishes on the social media page, and data items are characterized by the tags given to them. Similar to keyword-based approaches, the user is then given recommendations for data items from other domains that received similar tags as those that the user often uses when publishing data, e.g. the topics she is interested in. Apart from those two approaches, other relations, e.g. context factors such as time or the user's mood,

can also be used as an input for doing the recommendation task, similar to single-domain recommender systems as explained in earlier sections.

Collective and adaptive models

Apart from the data type that a recommender system is working on, cross-domain recommender systems can also be distinguished by the data sources and recommendation techniques used for building the user model [5]. Some of them use the data of one domain and try to give recommendations in only one other domain (*adaptive* models); others are *collective* models which use the data of several domains and can also make joint recommendations for multiple domains. Examples of *content-based adaptive* recommender systems are for example Karminskas and Ricci's work on location-adapted music recommending using tags [178]. In their work, they matched social tags for the location where the user is currently located, and matched them with music tracks with similar tags, which are then recommended to the user. Similar work has been done by linking semantic-based knowledge of two domains together in the example of recommending music for places of interest [111]. In contrast to the former work, this approach was more generic and could be implemented for two arbitrary domains. The semantic knowledge was provided in the form of RDF models offered by the Linked Data[4] project, which offers a global data space connecting data from multiple domains like people, companies, books, films, television, music, statistical and scientific data, and reviews [38].

Social tags have also been used for *collective content-based* recommender systems, for example for completing form-based data (entered explicitly by the user on forms on social media websites) or tag-based user profiles, so that even for social media websites that are not often used and thus have only sparse information about the users can be given strong recommendations by assembling and transferring user models from other social web pages the user has used in the past [1]. Other works focused on social tags in multiple social networks, and used Wikipedia as a multi-domain model to do the subsequent semantic modeling of the user's interests [317].

On the side of *adaptive collaboration-based* recommenders, mostly ratings were used for matching the user to similar users, for example to recommend books that are rated as good by users that gave a similar rating to the same movies as the user in question [345]. This is especially helpful to solve the sparsity problem, e.g. if one domain exists with a high rating density, to transfer the ratings to another domain with a low rating density, and (for example) to recommend movies in such a domain where only sparse data is available, e.g. where no or only a few movie ratings are available [204]. The same can also be done for heterogeneous data, for example if both user ratings as well as "clicks", i.e. the number of views, exist and should both be used across several domains, by using a principled matrix-based transfer learning framework [243].

Similar to collective content-based recommender systems, there also exist approaches for *collective collaboration-based* recommender systems, which use rating matrices containing user ratings from multiple movie and book websites, and then use classification algorithms in order to find similar users with data present in the domain in question, so that this can be used for a recommendation [205]. Some approaches are even able to integrate binary ratings (i.e. "like" or "dislike") into numerical ratings (like a rating from 1 to 5), and to use both for recommending items in the target domain [243]. Some of the approaches also use a probabilistic-based approach which allows the knowledge to be adaptively transferred across the domains

[4] http://linkeddata.org (last accessed: 2020-03-09)

by deriving the correlation between the domains automatically, thus increasing the prediction precision [357].

Ubiquitous user modeling

Another possibility for performing user modeling across domains is described in the ubiquitous user modeling approach by Dominikus Heckmann [152] and was designed especially for ubiquitous systems. More specifically, the ubiquitous user model does *not* transfer knowledge about the user from one domain to the other, but rather keeps the domain information separate in order to create a combined user model out of the domain knowledge. The approach consists of several parts: The General User Model (GUMO) contains persistent properties of a user, like her personality and characteristics, demographic information, emotions and profession. Additionally, domain-dependent interests (for example in books or movies) can be added to the general user model. In addition to this model, SituationReports describe the current situation of the user as captured by a ubiquitous sensor using an ontology, for example when the user is stressed (captured by a heartbeat sensor) or when she is about to miss a flight and is under time pressure (captured by the airline website and location sensors). A SituationReport contains several blocks ("boxes") of information: first the MainPart box describing the subject (user) and the situation (for example "Peter is under time pressure"), the situation box denoting spatio-temporal information such as start, end, duration and location of the situation, the explanation box including the source of deduction for the situation, e.g. which sensor collected the information and the evidence for the situation, and a privacy box containing information about the owner of the data, with whom it might be shared and for what purpose. Using these two parts, it is possible to describe the user's characteristics in general (GUMO) as well as the current situation the user is in according to the sensors of the ubiquitous environment. The approach also contains a reasoner, which allows merging context data (e.g. situations) from different domains described in the ontology. The reasoner can automatically resolve conflicts, for example if different sensors give contradictory information (for example the heart beat sensor detects an increased heart rate and reports an increased stress level, while a video camera detects a relaxed walking style, leading to the opposite assumption), or if they report in a different semantical or syntactical style about the same situation. Based on the merged contextual information and the general user model, the reasoner can output a report about the user's needs in the current situation, allowing the design of a user interface which fits the user's individual needs in that situation, for example by simplyfying the navigation through the airport when the user is in a hurry.

So far, privacy settings and privacy policies have not been part of the cross-domain user modeling research. In a later section, we will observe how privacy settings from other domains can be used to predict privacy settings for the target domain. The presented approach is a collective approach, as it uses the privacy settings from *all* other domains used in the study that are suitable for a prediction. The approach is further based on collective filtering, as the regressors used will be trained by study participants from which we know the privacy settings in *all* domains used in the study. The approach will then recommend to a user privacy settings similar to those of other users who chose *similar* privacy settings in the other domains.

2.8 Discussion

After a review of existing approaches in the last sections, we will now discuss their weaknesses that are of concern for this thesis and that are addressed by the approaches presented in the remainder of the thesis. For this, we will discuss in particular the problems that are present currently in privacy management systems and privacy user interfaces trying to assist the user in reducing the target audience and thereby reducing unwanted data disclosures ("narrowcasting").

2.8.1 Privacy management systems

There are two major types of privacy management systems to support the user in doing narrowcasting. Each has its own strengths and flaws.

Rule-based privacy management systems decide whether or not to disclose data based on rule sets, which are either written by the developer and adapted by the user, or written from scratch by the user. Although rule-based systems make it possible to easily trace back the decisions of the privacy management system by inspecting the rules, creating the rule set implies an additional user burden. Additionally, creating these rule sets requires the user to have technical knowledge about how rules have to be specified, and which rules are required and useful in which situations. If the rules cannot be specified using natural language, the user furthermore has to get used to the formal language specification of the rules. Systems inferring the rules from natural language can introduce an additional source of errors when translating the rules from natural language to formal rules. However, the created rules consist of explicit user feedback and therefore reflect the actual user intentions.

Machine-learning based systems can be again divided into three different kinds of ML-based systems: unsupervised systems that recommend privacy settings without additional user interaction, supervised systems that are based completely on user feedback for the prediction, and semi-supervised that combine user feedback and unsupervised learning for their operation. Similar to rule-based systems, supervised and semi-supervised systems rely on user feedback and thus create an additional user burden. Furthermore, they require technical knowledge about the consequences of a disclosement from the user, as he has to decide whether to allow or deny access requests in some exemple cases. On the other hand, they receive only explicit feedback from the user, and can therefore ensure that their learning input corresponds to the actual privacy desires of the user. In contrast to that, unsupervised systems have the advantage that they do not imply an additional user burden. Nevertheless, they need another data source for training their predictors. Usually, unsupervised systems try to learn from earlier prvacy decisions of the user, for example the privacy settings chosen for existing social network posts of the user. However, studies in the past have shown an effect called the *privacy paradox*, meaning that users' privacy settings significantly differ from their actual privacy intentions. Therefore, systems learning implicitly from earlier privacy decisions suffer from the privacy paradox by learning and recommending privacy decisions which do not fully comply with the privacy settings actually desired by the user. Both variants mostly use only context factors like the occasion or group of recipients as an additional input for their prediction. A small number of approaches assign the user a user stereotype based on her personality as an additional input. However, using individual measures as an input for deriving highly adaptive privacy recommendations, where each user is given an individual recommendation based on the

individual measures rather than based on the user stereotype, has not been part of research so far.

Furthermore, rule-based and ML-based approaches in the literature only allow the recommendation of a binary disclosement decision for social network posts. e.g. to show or to hide the entire post. Fine-grained privacy recommendations which also take into account the option to hide only parts of the post have not been discussed so far.

Cross-domain recommendation systems have so far been used only outside of the privacy settings domain, for example for recommending products of one domain (for example books) using ratings from another domain (for example movies). Recommending privacy settings for a domain (for example social network posts) using privacy settings from other domains (for example smartphone app permission settings) has not been discussed so far.

A summary of the strengths and weaknesses of the mentioned approaches can be found in Figure 2.36. The first column denotes the amount of user burden introduced, followed by whether it is prone to the privacy paradox, i.e. whether the learning input correlates to privacy settings to be recommended, and whether it is suitable for lay users by regarding the level technical knowledge needed. Finally, the last three columns denote whether the publications in that field use individual measures for a personalized recommendation of privacy settings, whether they offer fine-grained privacy recommendations instead of a binary recommendation to disclose or not, and whether they allow a cross-domain privacy recommendation.

The privacy framework included in this thesis offers a recommendation of privacy settings based on individual measures (see Chapters 4 - 6) rather than the user's privacy decision in the past, and thereby is unaffected by the privacy paradox. The proposed approach does not require technical knowledge, as the individual measures can be gathered either using a standardized personality and privacy questionnaire asking for subjective situation ratings, or by automatically deriving the individual measures from text written by the user, for example social network content (see Chapter 3). The approach offers multiple privacy levels instead of a binary privacy recommendation for social network posts, location sharing privacy settings, and privacy settings for an intelligent retail store like Amazon Go. The approach discussed in Chapter 7 allows us to recommend privacy settings without individual measures, by using privacy settings from other domains as an input.

2.8.2 Privacy user interfaces

The radar metaphor has been shown to allow a better overview on the current privacy state and possible privacy threats, thereby also actively engaging users in adapting privacy settings. However, the space inside a radar interface is limited, thus limiting the possible applications so far. Approaches trying to use radar interfaces for audience selection suffer from the fact that the audience that can be displayed inside the UI is very limited. Especially in social networks, where the potential audience consists of hundreds or even thousands of users, this limitation makes it impossible to use radar interfaces in a realistic scenario or in-the-wild studies. Another drawback of the radar metaphor so far is the lack of ability to display more than one privacy policy at once. Especially in more complex environments, like intelligent retail scenarios, that involve multiple different parties ("stakeholders") like retailers, family and friends, or marketing agencies that are interested in the data, it is hard to get an overview on the total privacy state at a glance.

	User burden	Immune to privacy paradox	Technical knowledge needed	Uses individual measures	Offers fine-grained settings	Cross-domain recommen-dations
Rule-based	Very high	✔	Very high	✘	✘	✘
ML -unsupervised	Very low	✘	Very low	✘	✘	✘
- supervised	High	✔	High	✘	✘	✘
- semi-superv.	Medium	✘	High	✘	✘	✘
This thesis	Very low	✔	Very low	✔	✔	✔

FIGURE 2.36: Strengths and weaknesses of privacy management systems in related work.

In Chapter 8, we describe two improvements on the radar metaphor. First we will describe techniques that allow us to display a large number of items, for example potential recipients of a post, inside a radar-based UI. The results show that despite the large number of displayed items, the error rate is still significantly lower using our improved radar-based UI compared to a conventional list-based interface, as is used in social networks nowadays. Second, we will show a way of displaying multiple privacy policies, for example for multiple stakeholders, at a glance, using a three-dimensional *privacy pyramid*. Our study highlights that the privacy pyramid allows a better overview on the current privacy state and makes it possible to detect unusual and potentially harmful settings, whereas the actual review and adaptation task is performed better using a traditional list-based UI.

User interfaces for grouping items have been a topic of research multiple times in the past. However, all research was focused on increasing the usability of the UI, or tried to transfer the sorting task to other input devices like virtual reality. In Chapter 8, we will discuss how new interaction technologies like virtual reality can be used to not only increase usability, but also to make the task of social network friend grouping more enjoyable by introducing gaming elements. A study will show that depending on the design, the usability and also the enjoyability can be increased. However, the degree of gamification also has a negative effect on the error rate, so that a trade-off between enjoyability and quality of the results has to be made.

In-situ feedback has been used for notifying the user about potential privacy invasions, for example for notifying the user about a possible hacking attack on her shared locations, by detecting if the location is accessed at an unusually high rate within a short period of time. The usage of smartwatches and social network notifications to get in-situ feedback on privacy violations by viewing and rating new comments or likes, as well as its possible applications to adapt privacy settings, have not been part of research so far.

The next chapter will show how the individual measures, which are used for the individualized recommendation, can be derived from the user's written text. Afterwards we will discuss the proposed privacy management system for social network posts and location sharing based on individual factors, before generalizing the approach to other domains in Chapters 5 and 6. The cross-domain recommendation of privacy settings is presented afterwards in Chapter 7, whereas the aforementioned user interfaces are discussed in Chapter 8.

Chapter 3

Prediction of individual measures using written text

The work presented here is based on already-published research [263]. Apart from predicting the privacy settings for various domains, another problem is the collection of the individual measures. Whereas related work used context features as an input that can be captured using (for example) post meta-information like the topic, time and location of the post, the retrieval of individual features like the personality and privacy measures cannot be done without the support of the user. In the aforementioned studies, we always used questionnaires to derive the individual measures. However, this procedure requires some time from the user, which he might not want to invest. Related work has already shown that a part of the user individual measures, namely the big five personality traits, can also be derived without the user's assistance, by analyzing written text of the user, for example blog entries, social network posts, or emails. In this section, we will examine whether the used privacy measures can also be derived from written text.

Personality is an important factor in people's everyday lives, as it influences most aspects of life such as job success [28] and overall happiness [242]. Even susceptibility to some diseases somehow correlates with personality [339]. But in the modern connected world, with an increasing amount of data stored for each person, privacy is also an aspect that is gaining importance in our life and the life of future generations. Personality has been used for many different use cases, from targeted advertisements on social network platforms [64, 70] through optimizing the songs played in a personalized online music stream [113] to a content filter for hotel ratings on TripAdvisor[278]. Although the individual measures can be used in many applications, the additional effort to fill in questionnaires with more than 100 questions leads users to avoid such recommender systems, even though they could benefit from their usage. Researchers have therefore searched for different ways to circumvent this problem: Most of them use publicly available data like social network entries or language features extracted out of the user's posts to perform a prediction of the personality traits.

However, the use of written text to predict a user's privacy measures has, to the best of our knowledge, not been examined so far. Within this section, we will therefore try to shed light on this topic, to be more precise, we attempt to answer the following questions:

1. Is there a correlation between profile information/writing style and privacy measures?
2. How precisely can the privacy measures be predicted, compared to predicting personality measures?
3. Which data set should be used (Facebook or Twitter posts, or profile information)?
4. If the personality traits are available, can they be used to predict the privacy measures? How good is the prediction compared to using social network content?

Furthermore, we replicate the results of current literature [104] with a reasonably large data set for Twitter users. We performed an online study, where we captured both the privacy and personality measures using traditional questionaires, as well as data from three different text sources that are potential candidates for input features for the prediction. Those sources are user posts from the Twitter and Facebook social networks, as well as the user's profile information, like workplaces he entered, number of friends, his hometown or his marital status. In a first step, we compute a correlation between the three data sets and the privacy/personality measures to observe whether they associate with each other. Based on the findings, we continue with a subsequent regression analysis to determine *how precise* a prediction of the individual measures can be, without further complex optimizations by a language processing expert. The paper ends with a discussion including guidelines describing the adequate source for the prediction of the different individual measures.

The results indicate that the privacy measures *can* be predicted significantly better than random, where the optimal source for the prediction is the user's posts from either Facebook or Twitter. The precision is similar to the prediction of personality traits. Personality traits can also be used as a prediction input, although language features provide the highest precision.

3.1 User study and correlation analysis

As stated before, the goal of the study was to find out whether individual features, especially the privacy measures, can be derived from a user's writing style on social networks or by inspecting the user's personal profile (like workplaces, hometown, number of friends etc.). In addition to the privacy measures, we also recorded the personality measures using a 44-item version of the big five personal inventory[172]. To be more specific, we captured the privacy measures using the IUIPC[1][221] questionnaire and the Westin Privacy Index [195] (see Chapter 2). The social network posts have been analyzed using the LIWC[2] 2015 text analysis software. LIWC is simple software to count text occurences belonging to certain categories. For this purpose, the software first performs a so-called *stemming* procedure, where each

[1]Internet Users' Information Privacy Concerns
[2]Linguistic Inquiry and Word Count

word is reduced to its word stem, for example by removing the declension (e.g. "houses" is replaced by "house"). After this, the software uses its built-in dictionary containing more than 80 categories of words to match each of the words in the given text. The software then outputs, for each of these categories, a percentage, describing how often a word of that category appears in the text. By that means, we can determine how much of the user's words belong to the category "sports", how often he uses neurotic-seeming adjectives, or how many exclamation marks can typically be found inside the user's texts, for example. For predicting personality, related work [132] has shown that it's actually *not the content* of the profile information items in social networks that correlates with a user's privacy measures, but rather *whether* the user decided to fill out and share a profile item. In addition to the language features, we therefore checked in our study whether the different fields of the "About" page of a user were filled out (later called *profile features*) rather than extracting the actual content. The next section will describe the procedure of extracting information, as well as the study procedure, in more detail.

3.2 Methodology

3.2.1 Online questionnaire

The study was implemented using LimeSurvey[3]; participants were recruited using Prolific Academic,[4]. We selected only participants that were actively using Facebook or Twitter. As the number of people using *both* Facebook and Twitter is only a small population, we split the study into *two* different studies to avoid a biasing of our results. In the first study, we required participants to be active Facebook users (using Twitter was optional), whereas we required participants to be active Twitter users for the second study (Facebook usage was optional). Therefore the amount of participants and data sets for the Facebook and the Twitter data sets slightly differ ($n_{Facebook} = 104$, $n_{Twitter} = 109$). The results were later merged for the analysis.

We paid the participants a compensation of 1£upon successful participation. Similar to earlier studies, the compensation was paid only if the survey passed our plausibility check, where we checked the survey for completeness and also wether the control questions had been answered correctly. We therefore obtained 110 results for each study. We filtered out six participants in the Facebook group as their profile contained too few posts to do a correct language analysis (the software manual states that a text should contain at least 300 words). In the Twitter group, we had one participant that had too few posts; we therefore had 104 participants in the Facebook and 109 in the Twitter group. The ages of the participants ranged from 18 to 72 years (average 33.02, SD 10.94). Again, like in earlier studies, we had a diverse audience consisting of students, self-employed workers, employees, and also homemakers.

The survey contained two parts. In the first part, participants had to answer the privacy and personality questionnaires. For the second part, users had to add a test Facebook friend given by us to their friend list and enter their Facebook ID (first study) and/or Twitter screen name (second study). The participants were given a description of which data is accessed, and infored that data access and processing were done anonymously, as described in the next section. We store only the language features derived from the LIWC tool, instead of the actual social network posts, and check *whether* the profile items are filled out, without extracting the actual content.

[3] https://www.limesurvey.org, last accessed 09-07-2019
[4] https://www.prolific.ac/, last accessed 09-07-2019

At the end of the questionnaire, participants were given the opportunity to leave feedback and comments.

3.2.2 Analysis of the social network profiles

After all participants finished the survey, the answers from the study were stored in a csv file. We then used an automated python script to extract the user's posts and profile items using the Selenium web automation toolkit [5]. Selenium is a web automation tool that was originally made for automated website debugging. It allows one to remotely control a web browser instance, simulate mouse clicks and keyboard input, and retrieve the content of a webpage even when it is dynamically generated using JavaScript, as on Facebook. In contrast to the Facebook API which is used frequently in related work, we can access and extract all data items that are visible to one of the user's Facebook friends (just like the test user that the participant had to add during the study), rather than the small part of profile information items that is accessible using the API. The script that we wrote first opens the friend list of our test user that the participants had to befriend during the survey. It then opens the profile page of each friend (i.e. each of our study participants), traverses the sections of the "About" page and records the posts the user wrote (excluding posts that friends wrote on the user's timeline) for the analysis using LIWC. As described earlier, we only recorded whether the entries on the "About" page (for more details on which entries were recorded, see Table 3.1) were filled out and visible to friends or not. The data we extracted from the social network sites is directly piped into the LIWC application; we stored only the output of the LIWC tool, e.g. the percentages for the different word categories. After all of the test user's friends (which, again, corresponds to all study participants) were analyzed, they were automatically un-friended.

The procedure described in the last sections was reviewed and approved by the ethical review board of our institution.

3.3 Results

As described in the last section, we recorded three portions of social network data: language features from Facebook posts and from Twitter posts, and the amount of data provided on the personal Facebook profile page ("profile features"). In addition to this data, we have the answers to the privacy and personality questionnaires (IUIPC, Westin privacy scale and big five personal inventory). Given the shape of our data (not all data items are normal-distributed), we decided to use a non-parametric test for the correlation analysis. For this, we performed a Spearman correlation between each privacy/personality measure and the language/profile features. The first three subsections give an overview on the correlation coefficients and significances for each of the three data sources: Facebook language features, Twitter language features and profile features. In addition to the correlation between the three data sources and the personality/privacy measures, we are also interested in whether it is possible to use the *personality* measures to predict the *privacy* measures, which is discussed in the fourth subsection.

[5] http://www.seleniumhq.org//, last accessed 09-07-2017

Section	Observed fields
General profile information	number of status updates
Friends	number of friends
Work and Education	number of work entries, number of education entries
Places You've Lived	number of places
Contact and Basic Info	number of contact entries, number of basic information entries, birthday, gender, religion, political views
Family and Relationships	relationship status, number of family members
Life Events	number of life events
Photos	number of photos uploaded by the user, number of photos uploaded by friends, number of albums
Likes	total number of likes, number of movies, TV shows, music, books, sport teams/athletes liked
Events	number of events visited in the past
Reviews	total number of reviews

TABLE 3.1: Observed Facebook profile items. As discussed above, we either counted the number of visible entries, or *whether* an entry is visible to friends.

3.3.1 Profile features

Table 3.2 gives an overview on the significant correlations that we observed using only the *profile features* that have been extracted out of the participants' Facebook user profiles (e.g. number of friends, shared workplaces etc.). All correlations have been computed with n=104 data sets. The most interesting findings are discussed below.

Value pair	**rho**	**p**
control - religion	.223	.017
control - gender	.230	.019
control - political	.230	.019
awareness - photos	.262	.007
collection - places lived	-.229	.019
collection - lifeevents	-.234	.017
collection - number of friends	-.233	.018
openness - family members	.209	.033
conscientiousness - music likes	-.202	.040
conscientiousness - movie likes	-.234	.017
conscientiousness - book likes	-.245	.012
extraversion - likes	.392	<.001
extraversion - sportlikes	.276	.005
extraversion - check-ins	.355	<.001
extraversion - events	.212	.031
extraversion - reviews	.338	<.001
extraversion - number of friends	.234	.017

TABLE 3.2: Significant correlations using only *Profile features*.

The correlations state that extraversion is the easiest to predict using profile features, supporting the findings of various earlier works [104, 19]. Based on our findings, *extraverted* people have more friends on Facebook, attend more events and like to publish their current location using the check-in functionality. Furthermore, they like to share their views in reviews, and tell people what they like. *Conscientiousness* correlates negatively with published likes on books, movies and music titles, meaning that less conscientious users tend to either be more engaged in music, movies and books, or to be more willing to announce this on Facebook than others. Finally, the *openness* to new experience positively correlates with the number of family members published on Facebook. Besides the personality measures, the three privacy measures of the IUIPC questionnaire also correlate with some of the profile features: *Collection* negatively correlates with the number of friends, life events, and earlier residences published on Facebook. In other words, users who place value on knowing which data is collected therefore tend to publish less information about their past on Facebook. Interestingly, users who emphasize self-control over their data tend to publish more potentially controversial personal views on Facebook: The *control* measure positively correlates with the probability of publishing political and religious views.

3.3.2 Facebook language features

Following the Facebook profile features, we started analyzing the language features extracted out of the status updates on their profile page. Note that we filtered out status updates that were posted by other social network members (like the friends or friends of friends) instead of the actual user. For the sake of brevity, only the results with a strong statistical significance are shown in Table 3.3.

Value pair	rho	p
extraversion - clout	.348	<.001
extraversion - risk	-.262	.007
extraversion - drives	.320	.001
extraversion - affiliation	.391	<.001
extraversion - cogproc	-.254	.009
extraversion - filler	-.282	.004
extraversion - friend	.255	.009
extraversion - social	.255	.009
agreeableness - adverb	-.265	.007
agreeableness - percept	.266	.006
openness - relative	-.258	.008
control - health	.258	.008
control - body	.288	.003
awareness - bio	.258	.008
awareness - shehe	.293	.003

TABLE 3.3: Significant correlations using only *Facebook language features*.

Again, extraversion has the most correlations throughout the set of features. Extraverted people do not write about their cognitive processes (*cogproc*), but like to talk about *friends*, *affiliations* and *social* affairs. They tend to prefer *clout* speech, using for example more swear words than usual. People who are open to new experiences (*openness*) use a lot of *relative* pronouns. Interestingly, it seems that users who put emphasis on *control* over their own data seem to be very *health*- and *body*-oriented, writing a lot about healthy products, activities, and the state and progress of their health and body condition. The values further indicate that people wanting to be *aware* of their current privacy settings seem to be *social* and write a lot about third persons (*shehe*), like "she is sooo cute".

3.3.3 Twitter language features

Twitter language features have been analyzed the same way as described above. Again we concentrated only on highly significant results in Tables 3.4 and 3.5. Note that we used the 37 data sets from the first study and the 72 data sets of the second study to come to a total of n=109 datasets. Again we will discuss the most interesting findings; a complete list of LIWC language can be found in the LIWC manual[249].

It can clearly be seen that there are notably more correlations for the personality measures, supporting the findings of earlier research that also received significantly better predictions using language features from Twitter [104]. Neurotic users in particular use language that is said to be typical for this kind of personality trait [78]: Users with a high neuroticism value are often *angry* and use *swear* words often. This

Value pair	rho	p
extraversion - affiliation	.318	.001
agreeableness - tentat	-.286	.003
agreeableness - Exclam	.315	.001
neuroticism - cogproc	.257	.007
neuroticism - swear	.322	.001
neuroticism - sixLtr	-.256	.007
neuroticism - analytic	-.275	.004
neuroticism - anger	.366	<.001
neuroticism - conj	.276	.004
neuroticism - negemo	.365	<.001
neuroticism - differ	.256	.007
neuroticism - interrog	.309	.001
neuroticism - function	.289	.002
neuroticism - verb	.264	.005
neuroticism - AllPunc	-.341	<.001
neuroticism - ipron	.260	.006
neuroticism - affiliation	-.274	.004
conscientiousness - anger	-.288	.002
conscientiousness - Exclam	.266	.005
conscientiousness - interrog	-.247	.010
conscientiousness - tentat	-.336	<.001
conscientiousness - ipron	-.264	.006
conscientiousness - cause	-.254	.008
conscientiousness - cogproc	-.341	<.001
conscientiousness - swear	-.282	.003

TABLE 3.4: Significant correlations for the personality measures using only *Twitter language features*.

could be caused by negative feelings (*negemo*). On the other hand, people with high conscientiousness rarely use *swear* words, and do not show *angry* behavior. They are rather *tentative*. Extraverts often discuss their *affiliation* with others.

Although we have a lot of features for the prediction of personality, it is rather mixed for the privacy measures. On one hand, we have three highly significant correlations for the Westin privacy scale measure. Based on our records, privacy-aware persons are *social*, but also *power*-driven, *competitive* and like to take a *risk*. On the other hand, there are only four correlations for the three IUIPC privacy measures. Unlike the Facebook language features, which had more correlations with the *control* and *awareness* measures, we have three highly significant correlations for the *collection* measure. According to the results, users that appreciate having knowledge about which data is collected are mainly *power*-driven, perceived as less *authentic* and often talk about topics concerning their *home*, like household products, renovations or issues with the landlord. Lastly, privacy *awareness* is negatively correlated with the religion feature.

Value pair	rho	p
awareness - relig	-.257	.007
collection - authentic	-.260	.006
collection - home	-.256	.007
collection - power	.248	.009
privacy index - risk	.254	.008
privacy index - power	.275	.004
privacy index - social	.254	.008

TABLE 3.5: Significant correlations for the privacy measures using only *Twitter language features*.

3.3.4 Personality

As stated in the introduction, we are also interested in whether existing personality measures can be used to predict privacy measures. Remember that we did the study in two parts; for the first we required only a Facebook profile to participate in the study, whereas the second study required an active Twitter account. Nevertheless, both questionnaires started with the personality and privacy questionnaires in *both* studies, before we asked for Facebook or Twitter data. Therefore we can use the answers of all participants of the first *and* the second study. We had 104 participants in the Facebook study, of whom 28 also provided a Twitter account. For the second study, we invited 81 more Twitter users, to come to a similar total of 109 Twitter data sets. The correlations presented in Table 3.6 are therefore based on the answers of $104 + 81 = 185$ participants.

	Privacy Index		**Control**		**Awareness**		**Collection**	
	rho	*p*	*rho*	*p*	*rho*	*p*	*rho*	*p*
Extraversion	-.016	.823	-.020	.777	.042	.566	.052	.470
Agreeableness	-.155	.031	.201	.005	.242	.001	.093	.198
Conscientiousness	-.115	.112	.036	.617	.339	.000	.138	.056
Neuroticism	.192	.007	.102	.157	-.039	.592	-.061	.398
Openness	.090	.211	.145	.044	.181	.012	.177	.014

TABLE 3.6: Correlations between personality and privacy measures.

Neglecting the collection measure, each of the privacy scales has a highly significant correlation with one of the personality measures. Agreeableness, as the best predictor, delivers strong correlations for the control and awareness measure, and a significant correlation for the Westin privacy index. Extraversion, on the other hand, does not have any significant correlation with the privacy measures. In total, the highest correlation coefficients can be found for the awareness measure, with both agreeableness and conscientiousness.

3.4 Discussion of the correlation results and hypotheses for the regression analysis

Our results follow the line of recent related literature [104], where personality could be predicted best using *Twitter* language features. The writing style on *Facebook* posts also correlates with the personality traits, although there are significantly fewer features that correlate compared to the Twitter data set. Lastly the *profile features* supply the least number of correlations, indicating that a prediction using regression algorithms will also perform worse compared to the two other data sets. Whereas *Twitter* is clearly the best source for personality prediction, we cannot discern a single data source that is perfect to predict *all* privacy measures: *Profile features* again show the least correlations, with six significant and only one highly significant correlation (note that for the other two data sets, only highly significant correlations were mentioned). The *Facebook* data set, on one hand, seems to be best for predicting the control and awareness measures, whereas the *Twitter* dataset provides more highly significant correlations for predicting the Westin privacy index and the collection measure. If the personality traits are available, they could also be a good source for predicting the privacy measures, as the correlation measures have shown. Especially the privacy awareness, control and the Westin privacy index values seem to be well-predictable. Nevertheless, the number of features provided by the personality traits is small compared to the number of language features provided by LIWC. We therefore suspect personality traits to perform worse than the language feature-based prediction, but still around the level of profile data prediction.

The strong correlations indicate that it is possible to predict the personality as well as the privacy measures of a user with the aforementioned data sets. In the next step, we want to find out *whether* it is possible to do so using a linear regression, how *precise* the prediction is using the standard deviation, and how good the prediction is *compared to a baseline approach* that predicts a constant value without using any feature input, denoted by the R^2 value.

Based on the previous results, we expect the following results for the regression analysis:

Personality traits:

1. Regression based on the language feature leads to a lower standard error than with *profile features*
2. Regression with the *Twitter* language features delivers a lower standard error than with *Facebook* language features
3. The standard error for predicting the personality measures is lower than for the privacy measures

Privacy measures:

1. Both language feature sets allow a regression with a lower standard error than *profile features*
2. The *Twitter* dataset yields the best results for approximating the *Westin privacy index* and the *collection* measure, whereas the *Facebook* data set delivers a lower standard error for the *control* and *awareness* measures of the IUIPC privacy scale
3. Using personality traits as regression input to approximate privacy measures is possible, with a standard error at about the level of profile data

3.5 Regression analysis

Related literature has shown that there is no significant difference between multivariate and univariate regressors when trying to predict personality using language features [104]. We therefore decided to use a univariate regressor for our prediction. To be more precise, we used two different settings: First we performed a linear regression with *all* features as input values. As using all features can lead to an overfitted model (expressed by a low coefficent of determination), we also performed the linear regression with a subset of the features. To be more precise, we used the *backwards elimination* method of the SPSS regressor, which starts with all features included and iteratively removes them feature-by-feature to find an optimum set of features that provides a good prediction with the least amount of features. This procedure can often, but not always, find a better solution than including all features. Especially if a model is overfitted with all features included, the heuristic can often optimize the prediction [18]. The results with all features will be labeled *all features* in the results tables; the ones with the minimal feature set can be found in the row *selected features*.

The values for the prediction of the personality measures are shown in Table 3.7; the results of the regression with the IUIPC and Westin privacy scales can be found in Table 3.8. Similar to the studies in the last sections, we computed the standard error of the prediction against the correct result, as well as the coefficient of determination (R^2).

The standard error using the *profile data* is highest for both the personality (0.588 for conscientiousness to 0.769 for extraversion) as well as the privacy measures (0.633 for the Westin privacy index to 0.956 for the collection measure). The personality measures as a data source perform at a similar level but slightly worse in predicting privacy measures (0.666 for the privacy index to 1.09 for collection). Both Facebook and Twitter language features lead to a notably better precision for the individual measures, whereas Facebook language features are slightly better for personality (0.328 for openness to 0.499 for extraversion) than Twitter (0.381 for openness to 0.619 for neuroticism), except for agreeableness (0.439 on Facebook vs. 0.390 using Twitter). The IUIPC measures are predicted better using Twitter tweets (0.375 for awareness to 0.643 for collection) compared to Facebook (0.423 for awareness to 0.661 for collection). The conventional Westin privacy index, on the other hand, is predicted best using Facebook (0.330) instead of Twitter (0.453).

3.6 Discussion

3.6.1 Precision of the prediction in general

Taking a look at the prediction precision, we can see that personality traits can be predicted quite well. The best results for the personality measures that range on a scale from 1 to 5 could be achieved with the language features. Contrary to our expectations, Facebook language data performs slightly better than Twitter language features. Taking a closer look at the full correlation tables reveals that although there are more strong correlations within the Twitter data set, the correlation measures for the personality traits in general are higher in the Facebook dataset. The lower number of features as well as the smaller correlation scores lead the profile data (information extracted from the user's "about" page) to perform worse, but still with a standard error between 0.549 (agreeableness) and 0.769 (extraversion).

Measure	Selected features		All features	
	stderr	R^2	stderr	R^2
Profile data				
- Openness	.601	9.3	.649	-5.4
- Conscientiousness	.588	20.8	.617	13.0
- Extraversion	.769	27.4	.816	18.3
- Agreeableness	.549	21.9	.588	10.7
- Neuroticism	.714	14.8	.755	4.8
Facebook data				
- Openness	.328	73.0	.515	33.6
- Conscientiousness	.410	61.5	.626	10.5
- Extraversion	.499	69.4	.777	25.9
- Agreeableness	.439	49.9	.751	-46.0
- Neuroticism	.467	63.5	.723	12.8
Twitter data				
- Openness	.381	61.3	.666	26.0
- Conscientiousness	.473	62.6	.647	30.0
- Extraversion	.542	61.5	.783	19.8
- Agreeableness	.390	62.4	.580	16.9
- Neuroticism	.619	53.3	.874	7.1

TABLE 3.7: Regression analysis for the personality traits using either all features, or only the most significant ones.

The IUIPC privacy scales can also be best predicted using Twitter language features. Having in mind that IUIPC measures range from 1 to 7, the standard error is not notably larger compared to the personality measures. Unlike what we expected in the discussion of the correlation results, not only the collection, but also the two other IUIPC measures are better predicted using the Twitter data, whereas the privacy index is predicted best using Facebook language input. Also here, a closer look at all the correlation measures led to the assumption that when comparing *all* correlation measures, the coherence between Twitter and the privacy measures is in general higher than with Facebook language features. Lastly, the profile data again performs worse than the language features, and at about the same level or only slightly better than the personality traits.

3.6.2 Comparing personality prediction with related literature

There have been several attempts to predict personality using textual input, most recently by Farnadi et al. in 2017 [104], where the authors compared the prediction precision for the big five personality traits using different textual sources of the user. They compared the precision using either Facebook posts, Twitter tweets or Youtube comments using a large database of 3731 Facebook users, 404 YouTubers and 44 Twitter accounts. In contrast to our study, this related work therefore had a significantly larger database; furthermore, they reported only the non-adjusted R^2 values which are always equal to or larger than the *adjusted* R^2 that we reported in the results. We were able to outperform the precision of their prediction with our Facebook and Twitter features, indicating a better goodness of fit of our model. The R^2 of our regression always ranged between 49.9 and 73.0 for the Twitter or Facebook

Measure	Selected features		All features	
	stderr	R^2	**stderr**	R^2
Profile data				
- Control	.844	15.0	.894	4.7
- Awareness	.674	30.7	.718	21.4
- Collection	.956	18.2	1.05	0.7
- Privacy index	.633	7.1	.683	-8.2
Facebook data				
- Control	.635	51.9	.940	-5.2
- Awareness	.423	71.7	.705	24.2
- Collection	.661	60.9	1.08	-5.7
- Privacy index	.330	74.8	.507	40.4
Twitter data				
- Control	.518	61.5	.724	24.6
- Awareness	.375	74.4	.532	48.4
- Collection	.643	68.3	.897	38.3
- Privacy index	.453	55.4	.681	-0.9
Personality traits				
- Control	.852	6.7	.879	0.8
- Awareness	.769	10.9	.777	9.0
- Collection	1.09	3.8	1.10	2.8
- Privacy index	.666	4.6	.667	4.0

TABLE 3.8: Regression analysis for the privacy measures using either all features, or only the most significant ones.

language features, compared to values between 2.56 and 17.78 in the mentioned publication. Furthermore, we achieved a smaller standard error for the prediction using Facebook (0.328 to 0.499 in our study compared to 0.649 to 0.776 in their study). Their results showed a smaller error for the prediction using Twitter tweets, but as the authors of that publication also stated, this might be an artifact caused by the the low number of Twitter accounts in their study ($n = 44$). To give more detail, they had a standard error from 0.152 to 0.214, compared to 0.381 to 0.619 according to our results. The prediction of privacy measures has, to the best of our knowledge, not been a subject of research so far.

3.6.3 Size of the training set

As stated in the introduction, and following the line of the studies described in earlier sections, the goal of the study was to find out *whether* the mentioned sources are suitable for predicting personality and especially privacy measures, and to find a lower bound for the prediction precision without further optimization. The goal was *not* to determine how far the precision can be increased, for example by optimizing the machine learning algorithm for this specific kind of task, or by optimizing the word categories of LIWC with the help of a language expert. Our study finds that the mentioned sources do *have* a significant correlation with the individual measures, and that they can furthermore also be used to perform a prediction of those measures that is better than random. Nevertheless, the prediction can be further optimized, either by optimizing the prediction algorithm or the text analysis, or even by just increasing the size of the training set. We are therefore interested in whether

the prediction precision can be increased when implementing the prediction as a Facebook plugin for some thousands of users, or even if it was integrated by the providers in their website.

3.6.4 Guidelines for the design of a privacy prediction algorithm

The results indicate that all data sources examined in the study sets can be used to predict personality, as well as the privacy measures of a user. The profile items from the "About" page of a Facebook user allow a privacy prediction with an acceptable precision better than random and can therefore be used as a first starting point. However, Facebook or Twitter posts are a notably better source for the prediction and should therefore be used first, if available. For predicting the IUIPC privacy measures, Twitter tweets provide the best precision, whereas the older Westin privacy scale can be slightly better predicted with language features extracted out of Facebook posts. If reliable personality traits are available, that kind of information can also be used, allowing a prediction of the privacy traits in a similar precision as the profile data.

3.7 Conclusion

As we have seen in Chapter 2, personality and privacy measures have a large influence on users' life choices and are therefore used as an input for recommender systems in research. However, although users could profit from these recommender systems, most of them are not accepted by users, as they are not willing to take on the user burden that comes with filling out questionnaires that may contain more than 100 questions each. Research has already found out that personality can be derived by doing text analysis on documents, short posts or blog entries written by the user. However, the derivation of privacy measures has not been a part of research so far. We therefore conducted a user study involving more than 100 Facebook and Twitter users, and examined whether there is a correlation between the words used in their personal writing and their personality and privacy measures. The results indicate that *there is* a correlation between those data sets, and it can also be employed to perform a prediction of the individual measures using simple machine learning algorithms. We gave guidelines on which data source should be used depending on the required precision and individual measure to be predicted. Although we were able to show the suitability of those input features and gave a concrete guideline on how to select them, we cannot state *how good* such a prediction can be when it is further optimized, which we would like to do in future work.

Chapter 4

Predicting privacy settings using individual factors in the social web

The work presented in this chapter is based on already published research [269, 260]. The social web, especially social media websites, have plenty of privacy settings that a user should adapt to her personal needs in the best case. However, as stated in the introduction, users rarely adapt them. So far, research either concentrated on analyzing the user's privacy behavior in the past, or, if such data was not available, used context factors as an indication to predict privacy settings for the user. Furthermore, related literature concentrated on proposing a binary choice to the user, to either disclose or not disclose the post. As other studies have shown, users need a more fine-grained approach for privacy settings, that also allows them to publish only parts of a post or an aggregated location [246]. In this chapter, we will investigate whether individual factors, like personality and privacy attitudes, can be used to assist the user in chossing her privacy settings, involving multiple fine-grained privacy levels, using machine learning. When choosing how to record individual factors or input variables *for multiple domains* in general, and especially which individual factors should be recorded, a researcher can use two different strategies that are mutually exclusive. The first strategy targets generalizability. In this case, a researcher tries to use a questionnaire that is as generic as possible while still gathering enough domain-specific data for the targeted domains. There already exist several commonly accepted questionnaires capturing personality [78] or privacy attitudes [221] for that purpose. This strategy has the advantage that the answers to the questionnaire have to be selected only once, and can be reused for privacy prediction in other domains. On the other hand, the second strategy targets prediction precision. In this case, a questionnaire is used that is highly specialized towards one targeted domain, including domain-specific questions that do not apply to other domains. This approach can lead to an increased prediction precision, although parts of the questionnaire answers might be useless for a prediction in other domains. In the ideal case, such a questionnaire uses a generic part that can be re-used for other domains, extended by some specialized questions for the domain.

In the first study, we performed a detailed user study involving such an *extended specialized privacy questionnaire* as well as a *validation study* in order to check whether the concept works in general, how good the prediction precision can be with a specialized questionnaire, and whether it is accepted in a realistic scenario, involving the user's real Facebook posts. In the subsequent study, we will check whether the concept can be transferred to the location sharing domain using only the generic part of the specialized questionnaire, and how good a prediction precision can be using a generalized approach. The same will be done for other domains in the next chapter.

However, the increased prediction accuracy that can be reached using a specialized questionnaire for the different domains mentioned here, as well as the derivation of these additional privacy measures without user interaction, remains for future work.

4.1 Social media domain

Creating a system which is able to automatically predict a privacy setting for friend lists is difficult if a user has just joined a social network. At that moment, little data is available to the user and also the provider, which makes it difficult to infer privacy settings automatically. Related research indicates that every social network post needs its own privacy setting that restricts the audience based on the *topic* of the post. Most social networks only allow a binary decision, e.g. either to hide or to show the complete post. However, for some of the posts, one might accept sharing only parts of the post by removing pictures or videos included in the post, or excluding the comments. Therefore we introduce a more fine-grained privacy scale to express a user's privacy desire, offering more than two decisions. As a starting point for our research, we started with a five-point scale similar to a likert scale to let the user express, for each friend list, to what extent she is willing to disclose the post (1 = strongly prefer to disclose, 5 = strongly prefer not to disclose). The answers to these scales are later denoted as the "privacy settings", and the five items on the scale as "privacy levels". In later sections, we will demonstrate a possible implementation of such a five-level scale, and discuss the advantages of a fine-grained scaling.

This chapter concentrates on the derivation of the privacy settings, and does not include a UI or front-end for the proposed system, as would be needed for a field study. For future work, we envision a user interface which could look similar to the one displayed in Figure 4.1, where the fine-grained privacy settings for the different friend groups are proposed by software, based on the topic of the post. The post topic extracted by the software can be seen on the left side in Figure 4.1, the proposed privacy settings (denoted by pictograms, according to the possible implementation of privacy levels in Table 4.4) are displayed on the right side. The "more" button is used to show the remaining friend groups. When the mouse pointer is placed on a pictogram, an explanation tooltip pops up that shows which content would be hidden by the setting. By clicking on the edit button on the left side, it is possible to override the detected post topic by manual input. As stated in the introduction, such a privacy assistant alone does not completely solve the problem, as the privacy setting prediction is never 100% correct, and even a small number of wrong privacy settings lead to mistrust by the user and thus a denial of the approach.

We introduce a questionnaire that determines the privacy attitude of a user in an online social network with a focus on privacy towards other users in the network. We suggest a machine learning approach to automatically derive the privacy level for each group of users for a given post. The machine learning software takes the answers to this privacy questionnaire as an input to recommend privacy settings for each post, tailored to the user's privacy attitudes. We conducted three user studies, which will be described in the following sections.

First we conducted a pre-study with a small set of participants, in order to discover which user groups are typically addressed in a private social network, and which topics are most recently discussed within posts. Second, the main user study helped us to refine the questionnaire, to reduce the set of required questions for determining the privacy settings, and to train the machine learning approach. We

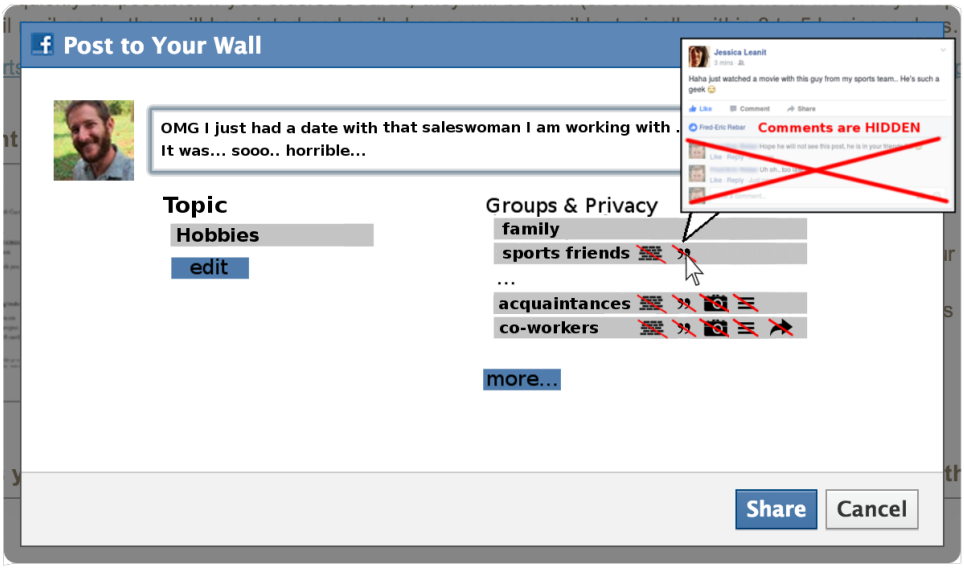


FIGURE 4.1: Envisioned user interface concept of a privacy setting prediction system.

verified the correctness and accuracy of the privacy setting prediction in a validation study.

4.1.1 Pre-study

The goal of the pre-study is to identify the most common topics of social network posts, and which groups of friends are typically addressed in such a network. Kelley et al. [184] conducted a user study on friend groups on facebook in 2011. As social networks have evolved in the last five years, we conducted a pre-study in order to replicate the results of this publication. We surveyed the participants about typical topics and friend groups that they personally encounter in everyday SN use.

The study was conducted as a qualitative online study using the software LimeSurvey[1]. We recruited the participants on Prolific Academic[2], which is an online recruiting platform like Amazon's Mechanical Turk. For the study, we selected only active SN users. The participants were paid of £2 if their participation was succesful, which means that their results were succesfully checked for plausibility.

In total, we surveyed 15 participants aged from 19 to 46 years (mean 30, SD 7.71). The audience of the study was very diverse: As participants we had, according to their own reports, students, self-employed workers, employees, and even a reverend. The questionnaire was divided into three sections, of which the first section inquired about the most relevant topics of posts, and the second part about friend lists in the social network they used. We did not ask for details on the members of the friend lists; we were just interested in labels describing them. The third section asked for concrete examples, regarding which groups a certain post should be shared with. The first two parts were structured the same way: First, we asked about high-level groups or topics that are often used. Such a high-level topic could be "sports", for example. Second, we requested more detailed topics or groups, or sub-groups of the aforementioned items. A sub-group for this example could be "winter sports". The subjects had only to imagine the post topics and friend lists they would use; we

[1] https://www.limesurvey.org (last accessed: 2020-03-09)
[2] https://www.prolific.ac/ (last accessed: 2020-03-09)

did not request them to copy actual friend lists or posts out of their social network into the questionnaire. The answers were given in free-text form. The subjects could enter as many topics or friend list names as they could think of. In the third part, we asked the participants to give some imaginary examples of posts and which groups they should be shared with.

We collected the topics and friend groups mentioned throughout the three questionnaire sections and clustered the results of the different questions manually using an axial coding approach [315]. We selected the friend lists that were mentioned nine or more times and topics mentioned ten or more times for the result set, as the amount of mentions again significantly dropped below that level. We found several groups that involved a special interest of the subject, like "Cincinnati Punk Rock Scene" or "Buy and Sell Groups". We clustered all these topics into the cluster *special interest group*. As this cluster contains a highly divergent group of users, we excluded it for the prediction in the main study. An overview of all mentioned topics, the different clusters, and the numbers of mentions can be found in Tables 4.1 and 4.2.

#	**topic**	*#mentions*
1	family	27
2	events	26
3	movies	24
4	politics	22
5	food	20
6	work	20
7	hobbies	14
8	travel	13
9	music	10
10	sports	10
11	feelings	7
12	achievements	7
13	news	7
14	tech stuff	5
15	health	1
16	religion	1

TABLE 4.1: Amount of mentions for all the mentioned post topics

group	#mentions
extended family	18
immediate family	17
work friends	16
close friends	12
special interest group	11
acquaintances	10
school/university friends	9
online friends	5
sports friends	2
not assigned	5

TABLE 4.2: Amount of mentions for the friend groups.

The results support parts of the results of the former study by Kelley et al.; most of the clusters observed in their study can be matched to the friend groups of our study. Table 4.3 displays the matching between the two sets of friend groups from both papers. All other groups listed in Table 4.2 could not be matched.

Kelley et al.	Our grouping
close friends	close friends
work	work friends
college / other education	school / university friends
family	immediate / extended family

TABLE 4.3: Matching of friend groups between Kelley et al. and the pre-study

4.1.2 Main study

The main study had two goals: First we wanted to refine the privacy questionnaire to reduce the number of necessary questions to a convenient amount. Second, the results from the study should serve as a training set for the prediction technique, which predicts a privacy setting for a new post, for each group of users (discovered in the pre-study), depending on the topic of the post.

For each of the five privacy levels, we gave the participants of our study a description regarding how the privacy levels are implemented, i.e. which parts of the post will be shown and which parts will be hidden, depending on the privacy level:

- Strongly like (level 1): The post is fully disclosed, and is actively brought to the attention of the receiving group (e.g. shown on their wall)
- Like (level 2): The post is fully disclosed, but not actively brought to attention (e.g. hidden on their wall; users can only see it if they visit your profile)
- Neither (level 3): The post is only partially disclosed (e.g. comments are hidden, or only textual information is shown without images or videos)
- Dislike (level 4): The post is not disclosed to the group (e.g. do not let members of the group see the post), but there are no additional actions to make sure the group cannot receive the information

- Strongly dislike (level 5): The post is not disclosed to the group, and there are additional actions to make sure the group cannot receive the information (e.g. if a direct friend of the group comments on the post or reshares it, it is hidden from the group)

What these implementations do in detail is map a privacy level to a set of binary settings for the different functionalities of Facebook, such as whether the comments should be hidden, the post should not be displayed on the timeline of a friend, or the post should be hidden completely. Table 4.4 shows an overview of the mapping of the privacy scale to social network functionalities regarding this post. I will further discuss the opportunities and limitations of the proposed privacy levels in the discussion section.

Functionality	**Privacy Level**				
	1	*2*	*3*	*4*	*5*
Shown on wall	X				
Comments shown	X	X			
Image content shown	X	X			
Post visible	X	X	X		
Visible if reshared	X	X	X	X	

TABLE 4.4: Overview of mapping from privacy level to SN functionalities

Like the pre-study, the main study was conducted as an online study, whereby recruiting was done using *Prolific Academic*. Notably, we took care that no participants from the pre-study were invited to the main study. In total, we surveyed 107 participants, of which 100 produced usable results. We omitted data sets which were either incomplete, or implausible according to the answers to the control questions. The participants were paid £2 upon successful participation after a plausibility check. Taking the survey took on average 18 minutes. The participant age ranged from 18 to 55 years (mean 29), again close to the age distribution of a social network site.

The study procedure was divided into two phases: First, we posed the questions of the privacy questionnaire (see left side of Table 4.6 in the Appendix). The second

	family	*events*	*movies*	*politics*	*food*	*work*	*hobbies*	*travel*	*sports*
sports friends	4.91	4.24	4.01	4.39	4.1	4.35	3.82	3.99	3.64
acquaintances	3.99	3.02	2.74	3.61	2.71	3.37	2.7	2.8	3.33
online friends	3.87	3.12	2.66	3.34	2.67	3.41	2.68	2.82	3.29
close friends	2.61	1.8	2.0	2.79	2.04	2.38	1.8	1.73	2.73
work	4.03	3.08	3.0	3.85	2.73	2.67	2.77	2.81	3.48
friends	3.06	2.17	2.32	3.04	2.18	2.66	2.03	2.01	3.0
immediate family	1.84	2.17	2.25	3.05	2.09	2.49	1.92	1.76	3.02
extended family	2.43	2.41	2.37	3.26	2.27	2.91	2.12	2.06	3.18
school/university	3.86	2.92	2.68	3.47	2.81	3.22	2.78	2.66	3.35
mean	3.4	2.77	2.67	3.42	2.62	3.05	2.51	2.52	3.22

TABLE 4.5: Mean privacy levels for each topic/friend list combination

	family	*events*	*movies*	*politics*	*food*	*work*	*hobbies*	*travel*	*sports*
sports friends	1.28	1.74	1.84	1.61	1.82	1.64	1.93	1.87	2.11
acquaintances	1.11	1.17	1.27	1.26	1.16	1.23	1.21	1.22	1.67
online friends	1.22	1.32	1.27	1.33	1.23	1.24	1.27	1.29	1.69
close friends	1.16	0.88	1.05	1.34	0.96	1.05	0.84	0.82	1.79
work	1.19	1.31	1.33	1.31	1.34	1.56	1.26	1.30	1.66
friends	1.15	0.81	1.07	1.25	0.92	0.98	0.77	0.74	1.68
immediate family	1.11	1.19	1.16	1.46	1.11	1.22	1.01	1.03	1.82
extended family	1.30	1.24	1.24	1.41	1.15	1.25	1.05	1.15	1.82
school/university	1.22	1.34	1.34	1.39	1.36	1.34	1.29	1.31	1.69

TABLE 4.6: Standard deviation for each topic/friend list combination

part asked for specific privacy settings for the post topic and friend lists that we derived in the pre-study, based on the privacy levels defined above.

The answers to the privacy questionnaire were given on a 5-point scale (1= strongly dislike, 5=strongly like). The answers to the second part (specification of the privacy settings for the combinations of friend list and topic) were also given on a five-point-scale (1 = strongly like the disclosure (= privacy level 1), 5 = strongly dislike the disclosure (= privacy level 5)) in the form of a table , where columns represent topics, and rows the friend lists. Below the table, the users were given an explanation on the meaning of the different privacy levels, as described earlier in this subsection.

Results

Table 4.5 shows the mean privacy levels for each friend list/topic combination, Table 4.6 the standard deviations. The topics form the columns, whereas the friend lists are represented by the rows in the table. Table 4.5 is colored according to the average privacy levels. The colors range from dark blue for a high mean privacy level (meaning "the least information is disclosed") to light blue for a low mean privacy level (meaning "the most information is disclosed"). A quick histogram analysis of the values for each combination of topic and friend group showed the data is *normally distributed*, which means that most of the users would choose a value near the mean of the recorded values. Therefore we can use this table to predict a post's privacy setting for every friend list in a social network, by using the mean privacy setting from the table.

Although this approach already offers different privacy levels based on context factors (here the friend group and the post topic), it does *not* provide an individualized privacy setting, that also takes the user's personality and privacy attitude into account. In the next subsection, we will propose a naive approach based on context factors as well as an approach that is using also such individual factors for a personalized privacy recommendation using machine learning.

Privacy setting prediction algorithms

We introduce *two* different mechanisms to predict the privacy setting for a given post and its topic: The *naive approach* takes the topic as an input and performs a simple table lookup on the data in Table 4.5 to predict the setting. When a prediction for a post about "food" is requested, the algorithm looks up the column "food" in Table 4.5 and uses the rows for a privacy prediction for each of the friend groups. Sports friends

are assigned level 4, acquaintances, work, school/unviersity and online friends receive level 3, and the remaining three friend groups are assigned privacy level 2 in this example. The second, more sophisticated technique uses *machine learning* to predict the privacy settings. The general approach for a Machine Learning prediction is to select a set of input questions, in this case the questions of the privacy questionnaire, which should be mapped to the privacy settings. For this kind of task, machine learning (ML) estimators are used. These estimators are trained with both the input and output questions and are able, if the right set of input questions are selected as features, to predict the output of new, unseen input questions.

There are a variety of possible ML estimators which are applicable for this kind of task, such as Lasso, SGD, and the Elastic Net regressor. We tried several approaches, and achieved the best results with a ridge regressor. To be precise, we used the ridge regression classifier of the scikit-learn[3] implementation with an alpha value of 2, which is the standard value and produced best results in our case. Scikit-learn is an open source implementation of various machine learning algorithms and can be obtained as a python library[4].

To choose the optimal set of input features, a naive brute-force method would need to examine all possible combinations of the privacy questionnaire questions, resulting in 2^{42} cycles of training and prediction. Because of the large runtime required for that computation, we had to apply a selection heuristic, which is a variation of the *wrapper subset selection (WSS)* [188] algorithm, which allows to reduce the amount of combinations to be tested by iteratively adding features and thereby minimizing the prediction error: This heuristic starts with an empty set of input features and adds new features successively. To decide which feature to select next, the ridge regressor is trained with the currently selected set of input features, including one of the remaining input features. We measured the performance of the currently selected feature set by training the ridge regressor with 75% of the initial dataset, followed by a prediction of privacy settings for the remaining 25% of the data set. After that we compared the prediction with the actual privacy settings given by the participants of the first study. The resulting mean squared error (MSE) is used as a score for the current set of input features.

The pseudocode to the WSS algorithm is nspired by earlier publications [188] and is shown in Figure 4.3. We start with two sets: the empty set of *current input features* and the set of *possible input features* which include all of the questions from the privacy questionnaire. In a first step, the algorithm iterates over all possible input features, selects the question with the lowest score/MSE, and starts over with the set of *current input features* which now includes the question with the lowest MSE. This procedure is continued as long as there are still features left in the set of *possible input features*.

The results of this part of the algorithm are subsets of input features with the specific score. Figure 4.2 shows a plot of such a result for the topic *politics*. The y-axis depicts the MSE for a given set of input features, while the x-axis depicts the number of input features contained in the specific set. As seen in Figure 4.2, the number of questions is not necessarily proportional to the value of the MSE. After adding the 13th question, the MSE error does not decrease further. In fact, it increases. We observed a similar pattern for all topics.

In its original version, the WSS algorithm would stop whenever there is no additional feature which decreases the MSE. On one hand, we cannot directly stop the

[3] http://scikit-learn.org/stable/modules/generated/sklearn.linear_model.Ridge.html (last accessed: 2020-03-09)

[4] http://scikit-learn.org/stable/ (last accessed: 2020-03-09)

optimization process whenever the MSE error increases in the next step, since as can be observed in Figure 4.2, there are multiple local minima of the MSE before the global minimum is reached. On the other hand, one of our goals is to keep the user burden at a minimum, which forces us to keep the number of required questions as small as possible. For all topics, we could observe that *if* there is a global minimum with more than 15 questions, this global minimum is not much better than the best local minimum with less than 15 questions. Therefore, we stopped the WSS algorithm whenever the set of questions contained 15 elements, and took the set of questions that correspond to the lowest minimum up to this point.

In addition to finding the optimal set of questions for predicting the sharing settings of a *single, specific topic*, we also computed a set of questions to predict an optimal setting for *all* of the possible topics. This set is later denoted as the *generic* set.

Apart from the problem of selecting the optimal set of input varibales, the second problem is the format of some of the variables. As stated in the background section, categorial variables are in general not suitable as an input for machine learning without further processing. However, both context features are categorial variables, which do not imply a certain order of the categories, that would allow us to transform the values to a number, similar to an ordinal scale. At a minimum, both context factors have a limited number of values. A technique that is often used in this case is the creation of binary *dummy variables*, one for each possible value of the variable. In the case of the *topic* variable, one would create nine dummy variables, one for each possible value of the variable ($topic_{familyaffairs}$, $topic_{movies}$...). To transform the original variable into the dummy variables, the dummy variable corresponding to the value of the original variable is set to 1; all other dummy variables are set to zero. These dummy variables can then be used as an input for a regression. However, this approach assumes an equal distance between all categories, which leads to a misinterpretation and a loss of precision in the regression algorithms. Another solution to the problem is to use advanced machine learning techniques that can automatically transform such categorial variables into ordinal variables or scales before usage.

A third new contribution that we want to propose here is an approach that we call *distributional machine learning*. In the training phase, this algorithm splits up the study data into several groups, one group for each possible combination of the categorial variables. In our experiment, this would lead to 9 (friend groups) * 9 (topics) = 81 different groups. In the next step, 81 regressors are trained using the 81 data sets, whereby the categorial variables are removed from the data set. In the prediction phase, the algorithm picks the correct regressor according to the given combination of context factors in the prediction request, and does the prediction with the given remaining variables.

So to wrap up again, we had three different options for dealing with the variables:

1. transform to dummy variables
2. use a categorial regression
3. use distributional machine learning

After testing each of the options, we received the best results with the last option, distributed machine learning. However, the optimal regression algorithm and also the best option for transforming the input data heavily depends on the domain data,

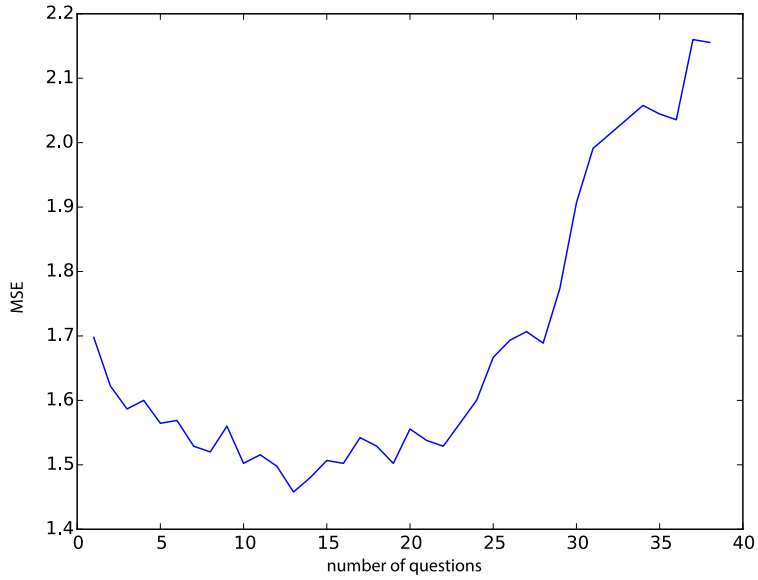


FIGURE 4.2: MSE for the topic "Movies/TV-Shows/Entertainment".

and therefore cannot be generalized across several domains, as will be shown for example in the location sharing domain later.

To use the estimators in the evaluation study, we trained the ridge regressors with 75% of the initial dataset, serialized the instances of the classifiers, and stored them in a Python module. In total, this procedure yielded one classifier for each topic, to predict the sharing settings for each of the nine friend lists identified in the main study.

Prediction algorithm results

The questions which were selected for each topic by the selection heuristic can be found in Tables A.1 and A.2 in the appendix. In total, 36 questions out of the initial set of 40 questions were used for the prediction. The set of questions selected to predict the whole set of topics is denoted by the column *generic*. The optimal prediction accuracy for a specific topic was achieved with six to fifteen questions, eleven on average. Although some questions were selected more frequently, we could not find any meaningful pattern. An overview on how often questions were selected is shown in Table 4.7.

The questions that were used most frequently by the WSS heuristic were: Q2 ("I feel out of touch when I haven't logged in to my social network for a while") with ten selections, as well as Q24 ("I want to keep my different social circles separate from each other on my social network") with nine, and Q7 ("I post very intimate things about myself on my social network"), which was selected eight times. These three questions have also been used for predicting the *generic* set, supporting the correctness of the selection algorithm.

The mean squared error of the prediction of the privacy levels against the test set is displayed in Table 4.8. We computed the mean squared error of the prediction separately for every topic set, as well as for the whole set of posts in the *generic* set. The best results were achieved for the topic *events* (MSE=1.33), whereas the

```
input_questions = [...]
current_qs = []

scores = []

while(len(input_questions) > 0 &&
          len(current_qs < 15)
{
        best_score = INT_MAX
        best_question = None

        for q in input_questions:
                s = score(current_qs + q)
                if best_score > s:
                        best_score = s
                        best_question = q

        scores.append(best_score)
        current_qs += q
        input_questions.remove(q)
}

global_minimum=min(scores)
return current_qs[:global_minimum]
```

FIGURE 4.3: WSS algorithm used for optimizing the questions.

question	*#selections*
Q 2	10
Q 24	9
Q 7	8
Q 10	7
Q 5	6
Q 9	6
Q 13	6
Q 15	5
Q 1	4
Q 16	4
Q 19	4
Q 28	4

TABLE 4.7: Number of selections for the most frequently selected questions. For the questionnaire-text, see Tables A.1 and A.2

topic	*mean squared error*
family affairs	1.48
events/plans	1.33
movies	1.46
politics	1.52
food	1.49
work	1.45
hobbies	1.48
travel	1.61
sports	1.61
generic	1.63

TABLE 4.8: Mean squared error for the prediction of the privacy level (from 1 to 5) for the selected topics in the main study.

travel and *sports* topics were hardest to predict, with a mean squared error of 1.61. Unsurprisingly, the mean squared error was highest with 1.63 for the generic set, containing all posts independent of topic.

4.1.3 Validation study

The privacy levels that were given by the subjects of the main study were based on imaginary posts, so our goal in the validation study was to test the prediction with actual posts out of the users' social network profile. We compare the users' privacy settings with the predictions of the naive and the ML approach. We did no direct comparison between the Facebook and the predicted *settings*, as Facebook's binary setting cannot be compared to our fine-grained five-point-scale. Therefore we let the user do a subjective comparison regarding the *satisfaction* with the predicted settings compared to their original privacy settings that were used when publishing the post on their social network profile.

As in the aforementioned studies, the validation was conducted as an online study, with recruitment by Prolific Academic. Participants of either of the former studies were excluded from the validation study. The number of participants totals

31 persons aged from 18 to 60 (mean 28). As in the two former studies, the participants were paid a compensation of £2 upon successful participation and after checking the submitted data for plausibility. The study started with the privacy attitude questionnaire, containing the questions derived from the main study. The privacy questionnaire is answered in the same 5-point scale that we used for the main study. As we also want to test for the RMSE of the topic-specific prediction, we took all 36 questions that were selected by the machine learning system.

The second part is divided into several phases. Figure 4.4 shows the website of this second phase of the validation study; the numbers denote the order of steps a participant had to take. *The UI was created only for the purpose of the validation study, and is different from a possible UI that implements the approach on a social network site*. First, the subject is given a list of pre-defined topics and an interface to copy/paste one of her posts that she published in a social network in the past (1). The topic of the post is then selected by the user through a drop-down list (2). As soon as the topic is selected, the second half of the page is displayed: On the upper part, the participant is asked to enter the privacy setting she used on her social network (3) when publishing the post (e.g. share with friends, friends of friends, or publicly). As soon as this information was entered, our approach unveils the proposed group-wise privacy setting (score between 1 - 5) for each friend list (4), along with a description of the effects of each privacy setting level from level 1 to level 5. The subject is given the possibility to adapt the setting suggestion to her needs (5). On the bottom of the screen (6), the subject states whether she prefers the proposed setting (before user changes) or her own setting (5-point scale, 1=own setting, 5=our predicted setting).

The second part is repeated until at least ten topics have been entered into the questionnaire. The questionnaire ends with a question asking whether the subject would use a system like ours if it was integrated into Facebook (1= very unlikely, 5=very likely). In addition to the answers to the questionnaire questions, we recorded whether the user changed the proposed privacy settings, in order to compare the changed setting with the prediction.

Results

To ensure quality, we excluded persons who failed to answer the control questions correctly, as well as posts whose content did not match the topic the subjects entered into the system, or posts where we could not verify this fact (for example because the post was written in a language that we could not translate). The final dataset therefore consists of 230 annotated posts.

To repeat, for each post as displayed in Figure 4.4, we asked, if they had to choose between their Facebook settings and our predicted settings, which one they would prefer. For this question, we performed a T-test against a constant value of three. As displayed in Figure4.5 , participants significantly preferred the predicted setting to their own privacy setting (M=3.23, t=2.369, p=0.019).

The results of the closing question “If such a system were integrated into Facebook, would you use it?” are displayed in Figure 4.6. More than 67% of all participants would likely or very likely use our system.

The users had the possibility to adapt the proposed setting if they were not fully satisfied with it. The mean squared error of the predicted settings to the adapted setting is shown in Table 4.9 aligned with the MSE of the naive approach and the amount of posts for each category. The results of the aforementioned studies will be discussed at the end of the chapter together with the results of the location sharing approach.

topic	*MSE (naive)*	*MSE (ML)*	*# posts*
family	0.90	0.93	38
events	1.16	0.85	24
movies	0.98	0.26	23
politics	1.28	0.91	14
food	0.83	0.46	28
work	1.00	1.17	18
hobbies	0.95	0.86	29
travel	0.76	0.64	17
sports	1.66	0.6	22
generic	1.15	0.78	230

TABLE 4.9: Amount of posts and mean squared error for the selected topics in the validation study for the naive and the ML approach.

4.2 Location sharing domain

Although a lot of social network sites also enable the user to share the location with her friends, for example by attaching the current location to a post or by "checking in" at a location; the decision whether or not to share the location depends on other context factors, like the occasion or type of the event the user is currently involved in, or the time of day and day of the week when the location is shared [31]. The influence of the individual factors like personality and privacy measures can therefore be different from the preceding experiment as well. In this study, we therefore have the goal to find out which individual factors are of importance for this domain, which machine learning technique can lead to optimal results, and what precision a user can expect from such an approach in the location sharing domain.

Especially in recent years, since smartphones with integrated GPS tracking became more and more popular, sharing the current location also became more popular. Whereas only 11% of all users shared their location in 2013, this number increased to 70% just two years later. Similar to social networks, users tend to censor their posts or in this case, tend not to share their location at all, if the privacy setting mechanisms of the location sharing provider are too simple and do not allow them to tune the disclosure settings in a fine-grained way [31]. However, introducing a more complex privacy setting mechanism alone does not increase user acceptance or desire to share one's location, as those systems require a lot of time to be adapted to the user's needs and require some technical knowledge about the meaning and consequences of the different settings [281].

So far, social network and location sharing providers have not implemented any solution to this problem. However, research has already proposed several approaches to automatically *infer* the privacy settings based on either explicit user feedback or by analyzing sharing decisions made in the past [281]. However, this approach only works for users who have already used the system before. Other researchers focused on context factors that have an influence on the sharing decision. Whereas first studies found indications that the time and day of the week also have an influence on the privacy decision [31], later studies found that it is not exactly the time and day of the week that is the crucial factor, but rather the type of the event or occasion also plays an important role [73]. Apart from this factor, the person requesting the location has been found to be a significant factor for the sharing decision by

multiple studies [74, 31]. Although context factors have been part of research multiple times in the past, the influence of individual factors has not been examined so far.

Studies have also shown that most users wish to have more privacy options than just "disclose" or " not disclose", including obfuscated or abstracted versions of their location, for example only the street name or the current city [246]. Using this functionality, it is still possible to tell your friends that you are visiting your new date in the town nearby, without disclosing the home adress of that person. Some social network providers took up this feature request and gave the users the possibility to apply some of these location abstractions. Nevertheless, the choice of the best abstraction level is quite hard, especially for lay users, as they have to make a trade-off between privacy (for themselves) and usefulness of the shared location (for their friends). So far, social web sites as well as approaches from research do not offer any support for this fine-grained privacy decision.

Similar to the approach presented in the last section, we conducted a user study including 100 online participants to find *whether* and *which* individual factors have an influence on the privacy decision (e.g., whether there is a correlation between individual factors and privacy levels) and, in the next step, *how much the precision increases* using those factors compared to a random prediction. In our study, including *seven* distinct privacy levels (or "location abstraction levels"), we were able to improve the prediction precision by up to 20%. At the end of the section, we will furthermore give some guidelines on which individual factors should be used depending on the available individual factors and the tolerable additional effort for the user to fill out questionnaires.

4.2.1 User study

We used a three-step approach for the main study, which will be described in the next three subsections. First, we collected a so called *gold set*, i.e. a data set that contains both individual measures as well as location sharing privacy settings for a larger number of users. We gathered the gold set using an online study, where participants were asked to fill in several privacy and personality questionnaires, as well as their location sharing preferences for different combinations of context factors. Using the gold set, we first analyzed *whether* there are correlations between the individual measures, and whether they are suitable for a prediction. Finally, we performed a regression analysis using a cross-validation approach to find out how precise a prediction can be using a machine-learning approach. As described in the introduction, research has identified two main context factors that influence the location sharing privacy decision: first the requestor, and second the occasion or event when the location is shared. Locations are very often shared on social network sites like Facebook, Google+ or Foursquare, and the groups of recipients are typically similar to the user's friends in social media[5]; we therefore re-used the groups of recipients from the social media study and other related work from this domain [184] for the location sharing study. Similar to the social media study, we use several fine-grained disclosure levels, which also allows users to publish an abstracted location to some of the recipients. To be more precise, we used the seven abstraction levels offered by the Google Maps API for our study, as described in Table 4.10.

To reduce side effects, we captured the individual factors, such as personality and privacy attitudes, using a questionnaire. However, as we have seen in chapter

[5] https://www.thewindowsclub.com/oversharing-on-social-media (last accessed: 2020-03-09)

Privacy level	Displayed location
1 - Exact location	exact GPS location
2 - Street & city only	area of the whole street
3 - City only	city area
4 - Province only	area of the province
5 - Country only	area of the whole country
6 - Continent only	area of the continent
7 - No location	none

TABLE 4.10: Privacy levels used in the location sharing study.

3, these measures can also be derived from the user's written text, so this step could be skipped at the cost of prediction precision.

Like the social network study, this study was also conducted as an online study using Limesurvey[6], whereas recuriting was done with an online recruiting platform called Prolific Academic[7]. We allowed only participants into our study if they were sharing their location online or using their smartphone. Each participant needed on average 10 minutes to complete the questionnaire, and was paid £1 afterwards. The questionnaire included three control questions. The participants were only paid if each of the control questions was answered correctly and if the participant fulfilled the study requirements mentioned before. If one of the requirements was not fulfilled, the participant was automatically rejected by the recruiting platform and replaced by another. We therefore ended up with exactly 100 valid participants. The participants were aged between 19 and 65 (average 33.08, SD 9.14). The audience had very different occupations, from employees and self-employed persons, to students, to homemakers. 46 of the participants were female, and 54 male.

The survey shown to the participants consists of two parts. In the first part, we captured the individual measures of the participants. We used the big five personal inventory to capture the personality, and the IUIPC and the Westin privacy index questionnaires without any modification for measuring the privacy attitude of the participants. In the second part, the subjects had to enter their preferred privacy levels (e.g. location abstraction levels) for each context factor combination. In order to ensure that the results would be comparable between subjects, we *did not* use their own locations shared in the past as an example. Instead, we asked them for a privacy level for a hypothetical post like "Imagine you are at a music event and share your location with your sports friends. Which privacy level would you choose?". We provided an explanation for each privacy level similar to Table 4.10 at the bottom of the questionnaire page. As a reminder, we identified *two* most important context factors that we used for the study: nine different friend groups ("family affairs", "events", "movies", "politics", "food", "work", "hobbies", "travel" and "sports"), and eleven different occasions (see Table 4.10), resulting in 99 individual privacy levels for each participant. The online survey ended with a text box, in which participants could enter feedback or proposals for improvement.

[6] https://www.limesurvey.org (last accessed: 2020-03-09)
[7] https://www.prolific.ac (last accessed: 2020-03-09)

4.2.2 Correlation analysis

Before starting with the correlation analysis, we first evaluted whether the chosen abstraction levels were useful for the participants, by analyzing how often each of the given abstraction levels was used within the study. Each of the abstraction levels was used by some of the participants. The least frequently used abstraction level *Continent only* was used by 18% of the participants, in total for 0.57% of the settings of all participants. Interestingly, the two binary options *Exact location* and *No location*, which are the only two options in many location sharing services, were *not* chosen most frequently. The most often used abstraction level was *City only*, which was used by 93 participants, on average for 32.65% of all settings. *No location* was used for 22.53%, *Exact location* for 17.84%, and *Street only* for 15.95% of the settings. Interestingly, there were more participants who used *Exact location* (75 participants) or *Street only* (74 participants), than who used *No location* at least once, leading to the assumption that if a user uses *No location*, she tends to do it for a larger portion of her privacy settings. Lastly, *Province only* and *Country only* were used only for about 5% of the privacy settings, thereby becoming the second least frequently used abstraction levels before *Continent only*.

As stated earlier, we are also interested in whether we can support the findings of related work regarding the context factors used in our study, e.g., whether the two mentioned context factors (type of occupation and group of recipients) have an influence on the privacy setting. For performing the variance analysis, we first computed the privacy levels averaged over all occasions (avg_{rec}) for each group of recipients for the analysis of the context factor *recipients*, and vice versa for the context factor *occasion* (avg_{occ}) , as described in Figure 4.7. Afterwards, we performed a variance analysis comparing the settings for the different context factor instances (for example comparing all privacy levels for the different types of occasions). A Mauchly test on both averaged privacy levels showed that sphericity is not given for any of them ($p<0.001$ for recipients, $p<0.01$ for occasions); we therefore performed a Greenhouse-Geisser test. The results indicate that both context factors have a strong influence on the privacy decision, e.g., that the variance between the context factor instances is highly significant. Whereas the F-value for the group of recipients is already highly significant ($F_{8,1092} = 3.329, p = 0.001$), the F-value for the context factor *occasion* is even higher ($F_{10,890} = 66.865, p < 0.001$), leading to the assumption that, although both context factors have a strong influence, the occasion when the location is shared has the highest influence.

In the next step, we checked whether the individual factors that we recorded also have an influence on the privacy decision, and whether they should be used for a privacy-setting prediction using machine learning. Both the answers to the personality as well as to the privacy questionnaires contain ordinal data, which is also not normally distributed according to the F-tests that we conducted prior to the correlation analysis. The results presented in Table 4.11 are therefore computed using a Spearman correlation, which is the nonparametric equivalent of a Pearson correlation.

The largest correlation coefficients can be achieved using the IUIPC privacy measures, especially the *collection* ($\rho = 0.167, p < 0.001$) and *control* ($\rho = 0.167, p < 0.001$) measures, indicating that these privacy measures have in general a stronger influence on the privacy decision, compared to the user's personality. All privacy measures have a positive correlation, meaning a higher privacy demand according to the privacy measures also leads to stricter privacy settings. The measures from the conventional Westin privacy index also correlate significantly ($\rho = 0.024, p = 0.018$)

Individual measure	rho	p
openness	-.071	<.001
extraversion	-.068	<.001
conscientousness	.060	<.001
agreeableness	-.024	.019
neuroticism	.037	<.001
collection	.167	<.001
control	.106	<.001
awareness	.050	<.001
privacy_index	.024	.018

TABLE 4.11: Correlations between individual features and the average privacy levels avg_{all}.

with the chosen privacy levels, although the effect is not as strong as with the two aforementioned privacy measures. The five personality traits all have a significant correlation with the privacy levels; apart from the *agreeableness* measure, alls of them are also highly significant. Openness, extraversion and agreeableness have a negative correlation with the privacy levels, indicating that open and extraverted people in general use less strict privacy settings, and have a higher focus on sharing data with their friends, rather than protecting their privacy. In contrast to this, neuroticism and conscientiousness, which can be found more with intraverted people, leads to an increased strictness in privacy settings.

4.2.3 Correlation analysis discussion

Each of the abstraction levels that we offered to the participants was actually used in the study. Participants used the intermediate privacy level *City only* most of the time instead of one of the binary options (exact location/no location), highlighting the need for fine-grained privacy levels in the location sharing domain. Even the least frequently used abstraction level, *Continent only*, was still used by 18 participants, although only for a very small number of settings. Nonetheless, there seems to be a need for this option, which led us to the decision to include *all* abstraction levels in the regression analysis later.

The results from the variance analysis support the findings of related work regarding important location sharing context factors [246] and also the selection of context factors within the user study. Both context factors have been found to have a strong influence on the privacy levels. Interestingly, due to our analysis, the occasion seems to have a notably stronger influence on the settings in the location sharing domain than the group of recipients, which stands in contrast to findings from social networks [115]. This leads us to the assumption that, although the context factors are similar for both domains, things that you do in life, e.g. which events you attend and which ones you stay away from, are perceived as more private information than the things you are posting about, which means people feel more embarassed when others know about them attending events that do not fit the user's image, than when they post or talk about topics that do not fit their image.

According to the results, the personality is also a great indicator for predicting privacy settings. Within these traits, one can discriminate the privacy effects between people that like to be accompagnied by other people, and those that prefer to be on their own: People who are extraverted, open and agreeable typically have less

strict privacy settings than an average person. This group of people has a focus on being together, connecting with people, interacting with them, sharing their life with them; maybe they even want to be seen in a better light and want to show what they have that others do not. Therefore they like to share everything they experience in life with others. In order to do so, they prefer to share more information, at the cost of reduced privacy. On the other hand, people who are anxious, have a high neuroticism, and are conscientious, usually like to live on their own. They do not draw any energy from sharing their life with others; rather they perceive it as awkward if other persons know details about their life. In accordance with that, they also prefer to have stricter privacy, and share less about their life in the social web, especially when it comes to location sharing.

Nevertheless, our results indicate that the privacy measures are better than personality measures for the prediction. The Westin privacy index already has a significant correlation with the privacy levels; however, the IUIPC privacy measures *awareness, control and collection* have an even stronger correlation. This supports findings of related literature, where the Westin privacy categories have already been found to be too coarse-grained to allow a meaningful prediction of privacy behavior and privacy settings using these categories as an input [349]. Especially when users want to have high *control* over their data, and who should be able to see it or not according to the IUIPC questionnaire, users tend to choose stricter privacy settings . The same holds for the *collection* measures, which means that users want to have an overview on which personal data is collected, and by whom it is collected.

Based on the results from the correlation analysis, we pose the following hypotheses for the regression analysis:

H 1: Using only context factors, the *occasion* yields a better prediction precision than the *recipient*

H 2: Including the personality measures reduces the prediction error more than the context factors

H 3: The best prediction precision can be achieved using the IUIPC privacy measures

4.2.4 Regression analysis

Based on the results from the correlation analysis, we made the following design choices for the regression study:

- *Location abstraction levels*: The analysis has shown that all abstraction levels were used at least by 13 users. We therefore decided not to remove any of the abstraction levels.
- *Context factors*: The variance analysis has also shown that both context factors have a significant influence on the privacy decision; therefore, both context factors have been used for the regression analysis as well.
- *Individual factors*: The correlation analysis has shown a significant or highly significant correlation between all personality or privacy measures and the privacy settings. We therefore include all individual features in the regression study.

Prior to the execution of the regression analysis, the input data had to be prepared in order to be compatible with a regression algorithm. All privacy and personality measures have their origin in ordinal scales, and can therefore be directly used in some regression algorithms. However, both context features are again categorial variables, leading to a similar problem as in the social media experiment. As stated in the last section, using dummy variables assumes an equal distance between all categories, which often leads to a misinterpretation and a loss of precision in the regression algorithms. Unlike the social media domain, we received best results for the location sharing domain using an approach called *categorial regression* (CATREG) [226] based on a multivariate regression, that automatically transforms categorial variables into scales, and uses an optimization algorithm to find the best order for the categories, and also the distances between each of them. Typical parameters to tune the CATREG algorithm are the maximum number of steps to be used for optimizing the variable transformation (of categorial into ordinal variables) as well as the minimum distance ϵ that tells the optimization algorithm to stop prematurely if the optimization score increases by less than ϵ after a step.

For this analysis, we used the CATREG implementation of SPSS with a limit of 100,000 steps and $\epsilon = 0.00001$. The two context factors *occasion* and *recipient* were entered as nominal variables, the personality and privacy measures as ordinal variables. For each combination of input variables, we report the adjusted coefficient of determination ($adj.R^2$) and the apparent prediction error. The coefficient of determination (R^2) denotes how well the regression fits the actual regression curve and therefore denotes the *degree of fitness* of the regression. The values for the R^2 start from a value of 0, which means that the curve does not fit at all, up to 1, which means a perfect fit. However, although the R^2 gives a good overview on how well the regression fits the actual curve, it does not state whether the included coefficients are good coefficients for the prediction, or whether the selection of coefficients is optimal: If a new coefficient is added to the regression, it can only increase the fitness. In the worst case, if the new coefficient is useless, the R^2 just stays the same. In contrast to this, the *adjusted* R^2 also takes the number of coefficeints into account, and can also decrease when a new coefficient is added that does not signiciantly increase the fitness of the regression curve ("overfitting"). We therefore always report the *adjusted* R^2 throughout this experiment. The second important measure used in this study is the apparent prediction error (APE), which compares how good the prediction is compared to a random prediction. The values of the APE range from 0, meaning there is no prediction error at all, to 1 meaning the prediction is as good or bad as a random predictor.

The results can be found in Table 4.12. Already using only the most significant context factor due to the correlation analysis reduces the APE to 0.966, i.e. it makes the prediction better than random by 3.4%. Including the second context factor can again reduce the APE to a value of 0.964. Using individual factors, the apparent prediction error can be reduced significantly more. Adding the personality measures to the context factors reduces the APE to 0.90; using the IUIPC and Westin privacy measures reduces it to 0.898 and 0.897, respectively. Using the IUIPC privacy questionnaire, which is also part of the social network questionnaire, together with the most significant context factor *occasion* as an input, leads to a similar APE of 0.899. If all of the aforementioned individual measures and context factors can be used, the APE can be reduced to 0.808, meaning the prediction is about 20% better than random.

Input features	adjusted R^2	APE
Occasion	3.3	.966
All context factors	3.5	.964
Context + personality	9.8	.900
Context + IUIPC	10.1	.898
Context + privacy	10.2	.897
Occasion + IUIPC	10.0	.899
All	19.3	.808

TABLE 4.12: Coefficient of determination (R^2) and apparent prediction error (APE) for the regression analysis with the different groups of input variables.

4.3 Discussion

We tested the social network approach under *two* different conditions: In the *main study*, we asked the users for their settings preferences without giving them any suggestions, to check the prediction precision without influencing the result. In contrast to that, the goal of the *validation study* was to verify the PAPMAT system in a scenario which is as realistic as possible. We therefore gave the users settings proposed by the machine learning algorithm and let them adapt inappropriate settings, just as a typical workflow with PAPMAT would look.

In both scenarios, we were able to predict the privacy settings with a MSE lower than the naive method, although we did not exhaust all possibilities of the approach: First, we had only a small amount of training data, gathered from 100 participants. The machine learning system's MSE would be reduced if a larger base of training data were available. Second, we did not take advantage of the users' feedback on and changes to the predicted settings in the validation study to refine the prediction system. The approach is able to dynamically observe user changes to the prediction, and take this input as additional training data. This allows us to adapt the prediction mechanism to a specific user, and to increase the prediction accuracy. Despite that, users significantly preferred our privacy settings to their own settings, and were satisfied with the prediction. For only 86 out of 230 posts, subjects preferred their own setting to our version.

The machine learning approach performed better for all topics in the main study, and for eight of the ten topics in the validation study. Only two topics ("work" and "family") are predicted better by the naive approach, whereas mean squared errors for "family" are almost identical. Work posts have a divergent nature; the content and audience that should be able to see the post differ greatly depending on the occupation and work field of the user. A researcher would be more likely to want to share his work experiences with his community than a cleaner would. A naive approach that is independent of the questionnaire can achieve better results, although the mean squared error is still high compared to the other topics. In an advanced version of our concept, we might include the occupation and work field into the prediction, in order to ensure a lower MSE for this topic. The differences in mean squared errors for the topics that were better predicted by the ML approach are notably larger. For the topic "movies", we achieved a MSE of 0.26 for the ML approach, whereas the naive solution yielded a mean squared error of 0.98. Which algorithm is most suitable, depends on the effort the user wants to expend: On one hand, the results underline that the machine learning approach performs better than

the naive approach for most topics. On the other hand, the ML approach needs the answers to the privacy questionnaire in order to work, whereas the naive approach can function without additional data.

The location sharing approach was also evaluated in two steps: In the correlation analysis, we had the goal to find out *which* context and individual factors are suitable for a prediction, and *how precise* a prediction using these factors can be. The correlation analysis has shown that all of the included context and individual factors have a significant influence on the privacy settings, and should therefore be included in the prediction, whereas individual factors seem to be correlated more strongly than context factors which have been used in research so far. The regression analysis supports this hypothesis (H 2). Using only context factors, the APE is reduced more using the context factor *occasion* than using the *recipient*, supporting H 1. Finally, the best privacy questionnaire for the prediction was not the IUIPC but the Westin privacy index. We therefore have to partially reject H 3, as the IUIPC led to a smaller APE than the personality measures and since the APE using the Westin privacy index is 0.001 smaller than using the IUIPC.

Further research on appropriatness of privacy levels

A central aspect of both approaches is the fine-grained, five-point scale that we give the user to express his privacy preferences. Although it is a very common and broadly used technique to offer only a binary choice (like allow/deny), a user decision on privacy is in fact more than a binary decision. A SN user does not only think "I do not at all want my drinking buddies to know that I dance ballet as a hobby" or "I would really like my co-dancers to see the pictures of that ballet contest". There are also some groups of people, like university friends, where a user would say "It is OK if they see it. I do not want to completely cut them off from that information, but I also do not want to draw too much attention to it". In this case, the user would take some middle road, for example by sharing the post and the pictures with the university friends, but hiding them from their timelines (or hiding the image content), so they can only see them if they visit his personal profile, or by disallowing (possibly embarrassing) comments on the pictures. Similarly, according to our study, users want to share only the city they are currently in, without disclosing too many details and thereby reducing the risk of being stalked.

In Facebook and most approaches in related work, the user had to set these "medium" sharing options manually, as there is no such option between disclosing or not. In our case, the user would set an intermediate privacy level, and the corresponding privacy settings as shown in Tables 4.4 and 4.10 would be applied automatically. In order to give the participants of the studies an idea of what each of the five privacy levels on our scale are meant to express, and to form a more realistic and meaningful application which is understandable for the subjects, we gave examples of possible implementations as described in the studies. Nevertheless, as we did not conduct a study to determine which implementation yields optimal results. We therefore plan to explore the mental model of the users regarding the privacy levels and which privacy settings they map to in a future user study.

Earlier studies in location sharing have already found indications that users have a need for a fine-grained location sharing privacy setting, involving more options than just to share or not to share [246]. Our study supports these findings: *All* location abstraction levels that we offered in the study and that can also be implemented using the Google Maps API were used throughout the study. Even *continent only*,

used least, was still used by 18 of the 100 participants at least for one privacy setting. Even more interesting, the most frequently used privacy level was one of the abstraction levels. The binary choices *exact location* and *no location* were only used second and third most often. We therefore suggest that location abstraction levels should be made available whenever users have to choose their location. Whether all abstraction levels that we offered in the study are needed, whether the set can be reduced while still maintaining the same user satisfaction, or whether other abstraction levels should be included, should be considered in future work.

Context factors or individual factors for location sharing?

The most important finding from the regression analysis of the location sharing privacy levels is that, in contrast to the approach of using context factors for the privacy settings prediction, this approach is only the second choice according to the study results. Whereas using the two most recognized context factors in the literature can only reduce the APE by 3.6% for the fine-grained privacy settings, adding the individual features according to our approach can again decrease the APE by about 16% down to 0.808. According to the results, the best way is therefore not to use context factors, but to capture the individual factors by either using questionnaires or by deriving them automatically. However, if context factors are already available, these can be included for the prediction as well.

Advantages and disadvantages of the privacy questionnaires

The usage of a privacy questionnaire seems like more of a user burden at a first glance, compared to related approaches that get along without initial user input. An important difference from techniques that use available information to provide a prediction without user feedback is that the answers to our privacy questionnaire are *explicit* user feedback. We can be sure that the answers that are later used to offer a prediction are in fact correct answers that reflect users' privacy attitudes. In contrast, implicit user feedback that is recorded throughout everyday social network or location sharing usage may contain faulty or misleading information due to the so-called privacy paradox [27], which can hamper the prediction process. This is aggravated by the fact that current social media websites support only a binary disclosure choice. As discussed above, this type of setting does not neccessarily reflect a user's privacy desire. Moreover, we cannot interpolate the missing information to map a binary scale to our five-point or seven-point privacy scales. A second advantage of the questionnaire is the higher generalizability of its answers. The questionnaire determines a general privacy attitude in the context of the social web. Its answers could further be used to predict privacy settings for other similar areas, like personal retail data or data sharing with mobile apps. The results of other related work as presented in former sections are bound to the specific application context, and cannot be used outside it.

The number of privacy scale questions for social network privacy settings can be further reduced

For the proposed set of topics, we used 36 questions in the questionnaire. Participants of the main study needed only 2–3 minutes for the privacy questionnaire, whereas they needed 15–20 minutes to set all the privacy settings. Aside from the fact that the amount of time needed for the settings is therefore significantly reduced,

participants often gave general study feedback that setting all the privacy settings was overly cumbersome and boring. Still, we can further decrease the amount of time needed for the initial questionnaire: If the MSE of the "generic" prediction is sufficient, only ten questions from the questionnaire have to be answered. Finally, as we have seen in chapter 3, it is also possible to derive the personality and privacy measures out of written text, allowing us to further reduce the user burden. The same technique could also be used to derive the measures of the social network privacy questionnaire.

4.3.1 Conclusion

Social network sites have expressive ways to adapt the privacy settings of a post in a granular way, but still lack a suitable user interface for setting them in a convenient and understandable way. We introduced a questionnaire to determine the privacy attitudes of a user in a social network context. We outlined two approaches, a naive approach and a machine learning approach, in order to propose privacy settings to the user, which are tailored to the user's personality and privacy desires as captured by a privacy questionnaire. For each post, the approaches create a distinct privacy setting for each friend list. Unlike former work, we use the post topic as additional input to the prediction, rely on explicit user input using a new privacy questionnaire to train the predictor, and allow more than a binary disclosure decision by offering five privacy levels to let the user express his sharing desires.

In a small pre-study, we observed which groups of friends usually exist in a social network, and which topics were most recently posted about. In the succeeding main user study, we captured how the answers to the privacy questionnaire can be used to predict the privacy settings of a user. We introduced a naive solution as well as a machine learning algorithm, and verified the results in a validation study, using posts from an operating social network site. The results show that users significantly prefer the derived setting to the setting they chose when publishing the post on the social network. The average mean squared error of the machine learning prediction compared to their actual desired setting was low, which led to a high satisfaction with the predicted settings. The machine learning approach performed notably better for most of the topics. More than two out of three participants claimed to be willing to use such a system, if it was integrated into the social network platform. We gave some ideas on how the MSE can be further improved, and outlined a design sketch on what the user interface for such a system could look like.

The two experiments have shown that it is possible to predict privacy settings for a user by using individual factors as an input for a prediction, rather than context factors. The validation study has shown that the approach also works in a realistic scenario that involves users' real social network posts, and that the approach is accepted and preferred by the majority of users. In the following chapters, we will take a look at how the prediction needs to be adapted in order to work for other domains as well. Unlike the approach presented just now, we will concentrate on determining the important individual factors that must be used for a prediction, and on calculating a lower bound for the prediction precision for these domains, using standard personality and privacy measures. Nevertheless, the prediction can still be optimized by using custom questionnaires tailored for the specific domain, or by optimizing the machine learning algorithms to the specific needs of the domain data, which remains for future work.

Similar to the social media domain, privacy in the location sharing domain suffers from poor user interfaces that do not include any assistance to help users choose

their privacy settings. Moreover, the available binary disclosure choice has been shown to be insufficient for the majority of users. So far, related work that makes it possible to assist users with their privacy settings, even if the user has just started to use the service, has concentrated on the usage of context factors like temporal factors, the occasion of the location sharing or the recipient of the location. In the study, we observed whether *individual factors*, like the personality or the privacy measures of a user, can also be used to further increase the prediction precision. Instead of a binary scale that is used by most related work, we allowed seven different location abstraction levels in our study.

The study results support the findings in related literature that context factors have a significant influence on the sharing decision. However, the results also show that the user's personality, and especially the privacy measures, have an even larger influence. The results also indicate that location abstraction levels, as we offered them in our study, are highly appreciated by users. The two binary choices were used for only 40% of the settings, whereas 60% were set to a location abstraction level that is often not offered by most research approaches and location sharing websites. However, there is still some work to do, such as confirmation of the chosen location abstraction levels, the possibility to assist with posts with multiple or ambiguous topics, and finally an in-the-wild study, where the approach is implemented for location sharing in a social media or location sharing website, so that we can measure whether such an assistance system is accepted by users, how good the precision is with a large-scale training set, and how often users use the system in their daily routine.

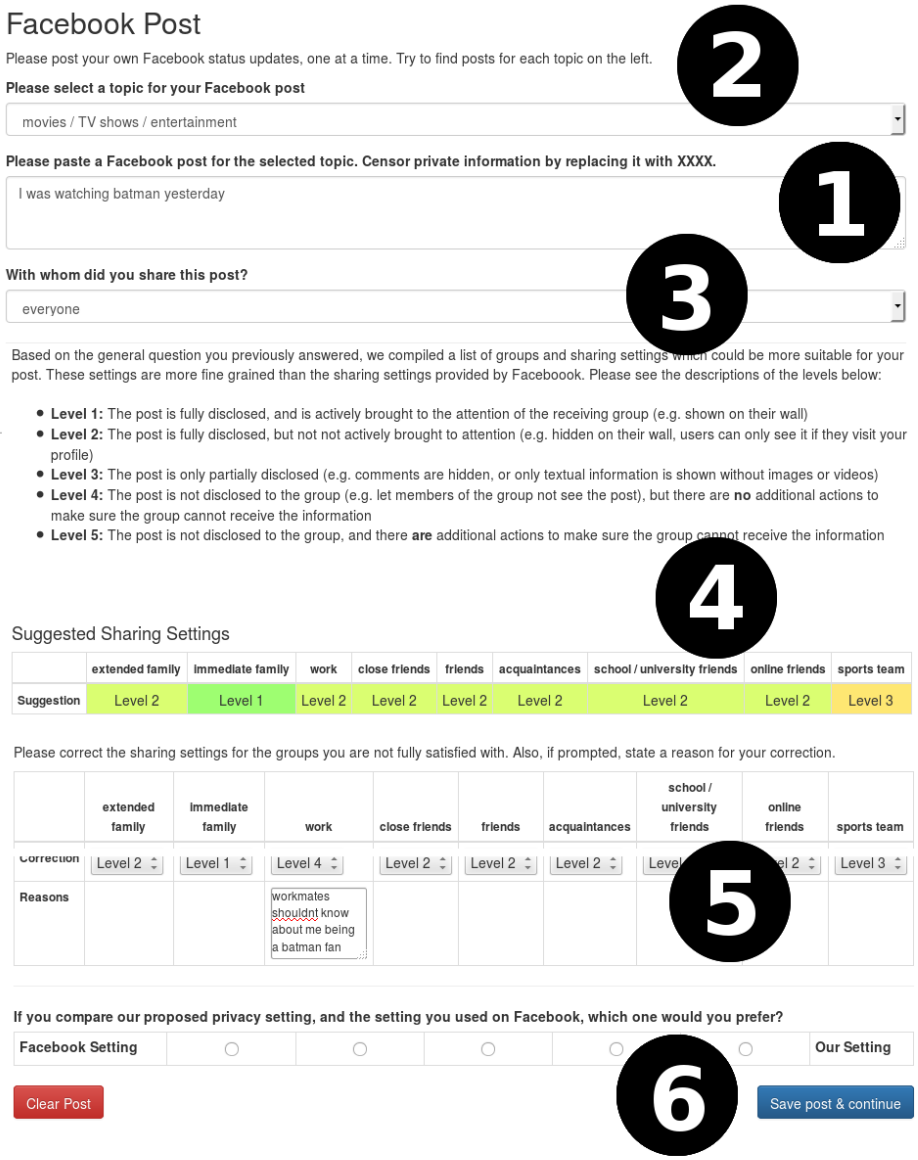


FIGURE 4.4: Screenshot of the webpage used for the second part of the **validation study**.

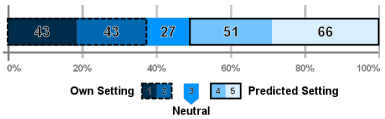


FIGURE 4.5: Settings preference for user's setting vs. the predicted setting.

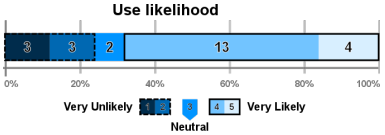


FIGURE 4.6: Use likelihood of the prediction system.

$$\forall rec \in recipients, \forall occ \in occasions : avg_{all} = \frac{\sum setting(occ, rec)}{|recipients| * |occasions|}$$

$$\forall rec \in recipients, occ \in occasions : avg_{occ} = \frac{\sum setting(occ, rec)}{|recipients|}$$

$$\forall occ \in occasions, rec \in recipients : avg_{rec} = \frac{\sum setting(occ, rec)}{|occasions|}$$

where $setting(occ, rec)$ denotes the privacy level for recipient *rec* and occasion *occ*.

FIGURE 4.7: Formulas to compute the overall average privacy level (avg_{all}) and the average privacy level for the two context factors "recipient" (avg_{rec}) and "occasion" (avg_{occ}) used in the location sharing study.

Chapter 5

Predicting privacy settings using individual and context factors – for smartphone app permissions

The work presented here is based on already-published research [264]. In Chapter 4, we presented our research on assisting users with their privacy settings in the social web, via the example of location sharing privacy settings and privacy settings for social network posts. The results show that, aside from the context factors, individual factors should also be used for predicting the user's privacy settings. Furthermore, fine-grained privacy settings that also allow abstraction levels, rather than just the option to disclose or not disclose the data, are needed. In this and the next chapter, we will show that a similar approach also works for other domains like mobile app settings and the data from an intelligent retail store. Although the problems and the approach are similar, each of those domains has its own challenges that have to be considered and solved during the implementation of the algorithm and the user study.

Smartphones, and especially their permission settings, have faced challenges since smartphone apps emerged more than ten years ago. The large number of apps and even larger number of permission settings introduces a massive privacy problem, as not only technical knowledge about app permissions is needed, but also a lot of time to tune *all* of the settings to a user's needs. As we will see throughout this chapter, our approach can also be adapted to work within this domain, using a different set of context factors (for example the app category or permission type) and individual factors for a prediction. However, the smartphone domain differs significantly from the aforementioned domains, as the number of apps is changing constantly, through new apps being installed, and old apps being uninstalled, raising a need for another workflow when using the privacy setting recommender system. Also unlike the social media domain, a permission can only be granted or denied, meaning there is a technical limitation to a binary privacy choice in this domain. Throughout the setion, I will describe and evaluate *two* different use-cases and approaches that are required for this special domain: one that is targeted towards a new user that has just bought and set up his new smartphone and wants to be guided in setting *all* of her privacy settings at once, and another approach that actively supports the user during his privacy setting process for a specific application that he has just newly installed.

In the early years of smartphones, the users did not have a possibility to change the permission settings of their installed apps. Each app came with a fixed set of permissions that the user had to confirm when installing the app. When the user did not want to grant one of the requested permissions, he had no other choice than

not installing the app. Even worse, studies found out that a lot of apps required permissions that they did not need for their intended use; they were often requested only for generating user profiles and targeted advertising [65]. Luckily, the smartphone users started to find a solution for the problem, which was first introduced as a hidden functionality in Android 3.4, and which became a permanent functionality available to users starting with Android 6.0. From this point on, users were able to allow or deny single permissions of an app, even after the app was installed and already being used.

However, although this functionality can in principle solve this privacy and security problem, OS manufacturers like Google and Apple did not introduce a powerful user interface that supports the user in finding the privacy settings for the apps that best fit the user's needs, and that explains the meaning of the permissions and possible onsequences when granting them to the app. According to studies, a typical user has on average 95 apps installed on his smartphone [240], each of them having five different permissions [239] leading to a vast amount of 475 permission which the user has first to understand, trade off the opportunities and risks, and then adapt according to his personality and privacy desire. Even if the user has the technical knowledge to do so, the large number of permissions is too burdensome, so that even knowledgeable users tend to adjust only a small portion of the permission settings, if any. Several studies have found out that most users either are unaware of the permissions they granted to the apps, or feel uncomfortable with their permission choice [109, 108, 147, 181]. Instead, most users rely on app ratings in the app store or play store, to decide which apps are "good" or "bad" and which of them they should install. However, privacy risks and data security are not part of the app rating in most cases [65]. Furthermore, research has already found out in several studies that the current way of displaying permissions and offering to grant or deny some of them, is not clear to users, and does not have the desired effect of informing them about potential risks and opportunities when granting some of the permissions [107, 109, 183].

Similar to other domains, research has tried to solve this problem by deriving privacy settings using large permission settings databases, containing the settings of millions of users, and using already-modified permission settings to propose permission settings for new apps [208], or by using context factors like the app category, permission type, or even the *purpose* of a permission, that has been automatically derived by analyzing libraries used by the app [209].

As we have seen in earlier chapters, individual features like personality and privacy measures have a significant impact on the desired privacy settings, sometimes even more than the context factors. Also, user behavior corresponds to personality: extraverted and open people tend to have more social network friends, publish more posts, and like other users' posts more than an average user [19]. People with a high conscientousness on the other hand, tend to submit less likes, and are rarely part of social network groups. The personality has also been shown to have an impact on user behavior in the mobile app domain: the user's personality has a significant influence on the apps installed on the smartphone [353]. WhatsApp users are typically extraverted, but also emotionally unstable [353], maybe because they are looking for a person to talk to or get some advice through an indirect online conversation rather than a face to face meeting with a friend. Twitter users are significantly less agreeable and more egocentric according to the study [353], just to mention two examples.

However, the usage of individual factors for assisting users in the mobile app domain has not been a subject of research so far. Similar to Chapter 4, therefore we

investigate in this chapter whether individual factors have a correlation to the permission settings, which of the individual factors have a significant influence, and which of them can be used for predicting the permission settings in a machine learning environment. To be more precise, we attempt to answer the following questions:

1. Do individual factors correlate with mobile app permission settings?
2. Which factors should be used as input for machine learning?
3. What prediction precision can be expected?
4. What might an approach, that supports a new user who wants to choose all of his privacy settings look like, and how can we also support an already existing user in choosing privacy settings for newly installed apps?

Toward solving these questions, we conducted an online study to capture individual factors as well as permission settings from 100 users; these were later used to analyze correlations and to train and evaluate a machine learning system. We propose two different approaches: one is an *a priori permission prediction* that offers a new user a guided process for adjusting the privacy settings for *all* of his privacy settings at once based on his individual factors; the other is the *dynamic permission setting prediction* that uses the individual factors to support the user actively while choosing the privacy settings on a newly installed app. The study results show that both the a priori permission prediction and the dynamic prediction improve upon the current standard and reduce the amount of user interaction needed.

5.1 User study

Similarly to the location sharing study, our target was to find out *whether* individual factors have a correlation to the permission settings, and which of the factors should be used for a machine learning prediction. It was *not* our goal to find out how precise such a prediction can be, if a very large data set of millions of users is used; we would like to postpone this question to future work, if our results indicate that individual factors are a fruitful source for a prediction. Furthermore, we did not do a separate study on a customized questionnaire for this domain, which could further improve the prediction precision. The results regarding the prediction precision presented here can therefore be seen as a lower bound for a prediction that might be possible with a customized questionnaire and a large data set. As input measures, we recorded the same individual factors that were also used in the two studies in the aforementioned domains, namely the big five personal inventory using the TIPI questionnaire [137] and the IUIPC privacy questionnaire [221]. Although it is already possible to derive the personality measures from a written text, which would also be the approach for an in-the-wild study using our prediction algorithm, we decided to record the personality measures using a questionnaire to reduce side effects for the study.

In addition to this generic part of the questionnaire, we also tried two domain-specific questions asking about the user's truthfulness when asked for sensitive data, and about experiences with privacy invasions in the past (see Table 5.1). To be more precise, we asked them how often (as a percentage) they give wrong information when asked by a website, and whether they have been target of a privacy invasion in the past, where data was misused or shared without their consent.

Label	Question
Falsify	Some websites ask you for personal information. When asked for such information, what percent of the time would you falsify the information?
Invasion	Have you ever been the target of a privacy invasion (e.g. your data was misused or shared without your knowledge)?

TABLE 5.1: Question text and label of the additional question set.

As we wanted to get answers that are as realistic as possible, we asked the users to give us the desired privacy settings for some of their apps, rather than asking for a hypothetical permission choice for several app categories. However, given the limited number of participants and study time, and to ensure comparability, we had to limit the number of apps covered by the questionnaire, as we will describe in the next section.

5.2 Methodology

The recruiting method of the study was similar to the studies described earlier: we had 100 participants that had been recruited using an online platform called Prolific Academic[1]. This time, we required that users owned and used an Android smartphone on a daily basis. The user had to have at least three new apps installed and used on a daily basis since the smartphone was bought. The privacy questionnaire included control questions, which allowed the recruiting platform to automatically discard answers that did not fulfill the quality standards or that did not fulfill one of the mentioned requirements. In that case, the study was ended prematurely for that participant, and the participant was not paid. If the study was completed succesfully, participants were granted 2£ for their participation.

The participants were aged between 18 and 61 years ($M = 30.13$, $SD = 8.53$). The occupations included students, employees, self-employed persons and also homemakers, just to mention the largest groups.

The survey that we gave to the participants again consisted of two parts, one asking about individual measures, another one asking about permission settings. The first part contained the aforementioned privacy and personality questionnaires. In the second part, we asked about permission settings that people would choose for a specific app that they were using on their smartphone. In that part, we *did not* ask how they chose their current permission settings for those apps, but how they *would* choose if they had the time and motivation to do so. The participants had to enter between three and ten apps in the questionnaire. We asked the participant, for each of the apps, to look up the individual permissions that were requested, which of these permissions the user would like to revoke, and the app's name, category and version (see Figure 5.1, upper left). It was only possible to mark "I would revoke the permission" if the permission was marked as used by the app before. As earlier research has shown, users struggle to find out which permissions are used by an app [147]. We therefore gave the participants clear step-by-step instructions on how to find the permissions for an app, as shown in Figure 5.1 (lower right). To make sure the users understood the instructions, we asked them to find out and enter the

[1] http://prolific.ac (last accessed: 2020-03-09)

FIGURE 5.1: Instructions on how to see the permissions requested by an app (lower right) and questionnaire page asking for used permissions of an app and permissions the user would revoke (upper left).

permissions used by a specific app (in our case Google Maps). Only if this task was done correctly users were allowed to proceed. If an app with the same version was entered by several participants, we used this opportunity to validate our data by checking whether they marked the same permissions used by the apps. In our study, this was always the case. At the end of the survey, users were allowed to enter comments and feedback in a free-text form.

5.3 Results

Altogether, 100 participants entered 876 apps into the questionnaire. To get an impression of how often a permission is denied to an app, and if so, how many permissions are denied, Figure 5.2 shows the number of apps together with the number of denied permissions.

As we can see, the majority of apps were granted all permissions that they requested (447 out of 876). For a significantly smaller number (147 apps), only one of the permissions was denied. Only a third of the apps had two or more denied permissions. Regarding the first part of the questionnaire, we computed the big five personality measures and the three IUIPC privacy measures as described in the literature [221, 137].

To get an impression of how likely it is for a certain user that a permission is denied, we computed a permission coefficient *comb* (see below) for each user, representing the likelihood that this permission is denied by this specific user. To do so, we counted how often a user denied access to a specific permission, and divided this by the number of the user's apps requesting the permission, e.g.
$\forall user \in participants, permission \in permissions$:
$comb(user, permission) = \frac{|denied(user,permission)|}{|requested(user,permission)|}$, where $requested(user, permission)$ means how many of the user's apps requested the permission, and $denied(user, permission)$

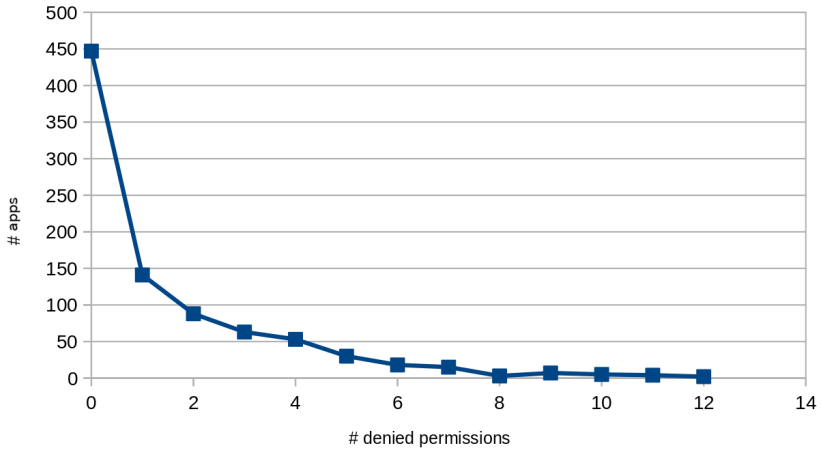


FIGURE 5.2: Comparison of apps and number of denied permissions.

how often the user denied this permission. By that means, we get a measure describing how likely a user will deny the permission, independent from the number of apps that he entered into the questionnaire. The value of these coefficients ranges from 0 (always granted) to 1 (always denied), and they are independent between participants. Table 5.2 shows the average permission coefficients for all permissions.

Taking a look at the permission coefficients, they seem to increase with the sensitivity of the permission. Accordingly, access to SMS, contacts and the current location is denied in every third case, which may also be because there are not many apps where a user can see a sufficient reason why the app might need this permission, apart from generating user profiles and creating spam. These three permissions are followed by access to stored photos, the camera, and the phone functionality which are also denied a bit less frequently than every third case. Access to the Bluetooth module and cellular information, on the other hand, is granted quite often, indicating that these permissions are perceived to be less harmful.

After computing the permission coefficients, the next step was to perform a correlation analysis to see *whether*, and how strongly the individual measures correlate with the chosen permission settings. As the data did not fulfill the requirement of sphericity according to a Mauchly test, we performed a Spearman correlation instead of a Pearson correlation. The results of this correlation analysis can be found in Figures 5.3 and 5.4. The individual measures are shown in the rows, whereas the app permissions are represented by the columns of the table. Highly significant correlations are marked in gray, significant correlations with a grayish background. Note that we let the participants enter apps that they installed and used on their smartphone regardless of the app category and permissions requested by these apps. Therefore, the number N of data sets for a specific permission differs greatly between permissions. Accordingly, the significance can also be higher in general for permissions having a large N. Therefore, the correlation coefficient is of more interest for deciding which individual measure should be used for a prediction.

The highest correlation coefficents can be found using the IUIPC privacy measures. Especially the collection measures have a high positive correlation with the permissions that are perceived to be sensitive, e.g. which have a high likelihood of being denied according to Table 5.2. All of the correlations are positive, meaning that users that have a high desire to know *which* data is collected also have a high

Permission	% denied
SMS	37.2
Contacts	36.3
Location	34.6
ID	31.0
Photos	30.9
Camera	28.9
Phone	28.6
Identity	26.6
Microphone	25.1
Purchase	18.4
Other	17.9
History	17.6
Calendar	16.7
Wifi	12.8
Cellular	9.5
Bluetooth	5.0

TABLE 5.2: Permission coefficents according to the study results for all permissions.

likelihood to deny most of the permissions. In total, the collection measure has a significant correlation with 11 of the 17 permissions; 6 of the correlations are even highly significant. The control measure has three significant correlations, namely with the purchase, contacts and location permissions. Also here, all correlations are positive, meaning that persons who want to have *control* over how and with whom their personal data is shared are also very likely to deny those three permissions. Finally, the awareness privacy measure has two significant correlations, with the contacts and photos permissions.

Although related work has often found correlations between *personality measures* and privacy choices, the effects that we observe in the results are mostly insiginficant. The *openness* personality trait leads to two significant correlations (namely for the location and SMS permissions). Interestingly, this correlation is negative, meaning that people open to new experiences are less likely to allow access to Location and to read incoming SMS. People with a high emotional stability are more likely to allow access to the Bluetooth module, and conscientious participants are less likely to allow access to the permission *other* according to our analysis. Although we did not achieve many significant correlations for the personality measures, the correlation coefficients still remain relatively medium-high, making it a potential candidate for the machine learning prediction. However, the custom questions did not produce any significant correlations; we therefore dropped those additional questions for the machine learning study and analysis.

The power of the analysis depends strongly on the permission type. We received more samples for permissions which are often used by apps (for example "contacts", "location", "photos", etc.), resulting in a reasonably high statistical power (> 0.8) for these permissions in most combinations. However, there are also other permissions which are rarely used by apps and are therefore rarely reported in the study. The "wearables" permission, for example, was used only by the apps of 17 participants. Therefore, the power of the correlation analysis goes down to 0.15 for the correlation between the "wearables" permission and the "collection" privacy measure. In order to achieve high statistical power (>0.8), approximately 236 samples would be needed

		purchase_comb	history_comb	cellular_comb	identity_comb	contacts_comb	calendar_comb	location_comb	sms_comb
control	Correlation Coefficient	,218*	-,004	,098	-,006	,202*	-,017	,236*	,050
	Sig. (2-tailed)	,038	,971	,587	,959	,040	,895	,015	,655
	N	91	82	33	89	104	62	105	84
awareness	Correlation Coefficient	,169	,170	,309	,064	,267**	,112	,178	,108
	Sig. (2-tailed)	,109	,127	,080	,554	,006	,387	,069	,326
	N	91	82	33	89	104	62	105	84
collection	Correlation Coefficient	,314**	,203	,370*	,314**	,344**	,324*	,297**	,209
	Sig. (2-tailed)	,002	,067	,034	,003	,000	,010	,002	,056
	N	91	82	33	89	104	62	105	84
Extraversion	Correlation Coefficient	-,043	,155	,148	,062	-,017	,121	-,124	,082
	Sig. (2-tailed)	,688	,163	,412	,562	,862	,349	,209	,460
	N	91	82	33	89	104	62	105	84
Agreeableness	Correlation Coefficient	,193	-,060	-,208	,020	-,032	,108	,117	-,012
	Sig. (2-tailed)	,066	,595	,245	,852	,744	,402	,235	,911
	N	91	82	33	89	104	62	105	84
Conscientousness	Correlation Coefficient	,097	-,088	-,106	-,047	,020	,009	,087	,036
	Sig. (2-tailed)	,360	,432	,557	,665	,842	,942	,375	,745
	N	91	82	33	89	104	62	105	84
Emotional_Stability	Correlation Coefficient	,161	,050	-,165	,014	,005	,115	,059	-,005
	Sig. (2-tailed)	,128	,656	,359	,896	,957	,374	,552	,966
	N	91	82	33	89	104	62	105	84
OpenExperiences	Correlation Coefficient	-,023	-,094	-,233	-,093	-,144	-,171	-,259**	-,229*
	Sig. (2-tailed)	,832	,400	,191	,386	,145	,184	,008	,036
	N	91	82	33	89	104	62	105	84
invasionfrequency	Correlation Coefficient	-,012	,076	,158	,158	,084	,160	,110	,126
	Sig. (2-tailed)	,918	,536	,433	,179	,438	,258	,310	,298
	N	81	68	27	74	87	52	87	70
falsify	Correlation Coefficient	,077	-,065	-,060	,028	,096	,181	,095	,065
	Sig. (2-tailed)	,495	,596	,766	,815	,377	,199	,383	,591
	N	81	68	27	74	87	52	87	70

FIGURE 5.3: Correlations between the privacy/personality questions and app permission settings.

for that correlation. Assuming that, similar to our study, only 17% of the participants report an app with this permission, about 1400 participants would be needed to achieve high statistical power for this analysis.

In addition to the correlations between individual measures and permission settings, we also analyzed the correlations between the permissions, which we will need later for the second use-case, where we help the user to adapt the permission settings for a new app by using the user's changes to the first permission entries to predict the needed changes for the remaining permissions. The results are shown in Tables 5.5 and 5.6. The correlations are notably stronger compared to the first correlation analysis. The correlation score ranges from 0.217 (Phone–Wifi) up to 0.859 for the combination cellular information – identity. Similar high correlations could be found between cellular information and purchase (r=0.81) as well as history (r=0.778). Almost all combinations have a significant or highly significant correlation; in the cases where they do not, it is very likely because the combination of those permissions is very rare, and therefore N is too low for the correlation to become significant.

5.4 A priori permission setting prediction

As the correlations analysis has shown, there are several individual measures which are suitable as an input for machine learning. Currently, the Android OS allows all permissions as a default, which goes hand in hand with the results in Figure 5.2. In line with related literature [209, 208], we therefore concentrated on the hard cases for the machine learning analysis, where at least one permission is denied. As stated in the background chapter, the choice of the machine learning algorithm first depends on the kind of data and kind of prediction to be performed. Unlike in the previous studies, we now have only two distinct choices (allow or deny), which results in a classification problem in machine learning. Typical algorithms that are used for

		phone_comb	photos_comb	camera_comb	micro_comb	wifi_comb	bluetooth_comb	wearables_comb	id_comb	other_comb
control	Correlation Coefficient	,085	,163	,167	,039	,017	,148	-,128	,050	-,020
	Sig. (2-tailed)	,468	,106	,101	,707	,871	,282	,624	,647	,852
	N	75	100	98	95	98	55	17	88	94
awareness	Correlation Coefficient	,106	,198*	,027	,126	,116	,129	-,182	,171	-,068
	Sig. (2-tailed)	,365	,049	,792	,225	,257	,349	,485	,112	,517
	N	75	100	98	95	98	55	17	88	94
collection	Correlation Coefficient	,163	,333**	,254*	,220*	,246*	,135	-,026	,333**	,083
	Sig. (2-tailed)	,161	,001	,012	,032	,015	,324	,922	,002	,427
	N	75	100	98	95	98	55	17	88	94
Extraversion	Correlation Coefficient	,040	-,088	-,022	,115	,078	,062	,077	,071	,141
	Sig. (2-tailed)	,733	,385	,829	,268	,446	,652	,768	,509	,175
	N	75	100	98	95	98	55	17	88	94
Agreeableness	Correlation Coefficient	,018	,108	,103	-,074	,015	,067	,181	-,083	,017
	Sig. (2-tailed)	,880	,286	,313	,475	,884	,629	,486	,441	,874
	N	75	100	98	95	98	55	17	88	94
Conscientousness	Correlation Coefficient	,012	,025	-,072	-,018	-,123	,079	-,104	-,067	-,20*
	Sig. (2-tailed)	,919	,806	,481	,863	,228	,568	,692	,536	,048
	N	75	100	98	95	98	55	17	88	94
Emotional_Stability	Correlation Coefficient	,006	,024	-,135	,022	,026	,351**	,130	,005	-,130
	Sig. (2-tailed)	,962	,813	,186	,835	,800	,009	,619	,965	,212
	N	75	100	98	95	98	55	17	88	94
OpenExperiences	Correlation Coefficient	-,201	-,039	-,162	-,116	,005	,029	,206	-,168	,031
	Sig. (2-tailed)	,084	,697	,112	,265	,957	,836	,427	,119	,764
	N	75	100	98	95	98	55	17	88	94
invasionfrequency	Correlation Coefficient	,118	,173	,002	,091	,035	-,058	,000	,045	-,055
	Sig. (2-tailed)	,364	,116	,989	,421	,759	,710	1,000	,708	,633
	N	61	84	83	81	80	43	13	71	79
falsify	Correlation Coefficient	,195	,148	,094	,000	,008	,022	-,428	,078	-,014
	Sig. (2-tailed)	,132	,180	,397	,999	,945	,888	,144	,517	,901
	N	61	84	83	81	80	43	13	71	79

FIGURE 5.4: Correlations between the privacy/personality questions and app permission settings, continued.

			Purchase	History	Cellular	Identity	Contacts	Calendar	Location	SMS
Spearman's rho	Purchase	Correlation Coefficient	1,000	,664**	,810**	,556**	,286**	.	,531**	,273
		Sig. (2-tailed)	.	,000	,000	,000	,005	.	,000	,064
		N	228	73	49	125	93	27	98	47
	History	Correlation Coefficient	,664**	1,000	,778**	,583**	,416**	,502**	,501**	,350**
		Sig. (2-tailed)	,000	.	,000	,000	,000	,000	,000	,000
		N	73	227	68	170	143	74	153	114
	Cellular	Correlation Coefficient	,810**	,778**	1,000	,859**	,397**	,379**	,135	,472**
		Sig. (2-tailed)	,000	,000	.	,000	,001	,006	,300	,001
		N	49	68	105	65	71	52	61	48
	Identity	Correlation Coefficient	,556**	,583**	,859**	1,000	,556**	,344**	,455**	,419**
		Sig. (2-tailed)	,000	,000	,000	.	,000	,001	,000	,000
		N	125	170	65	433	261	93	295	157
	Contacts	Correlation Coefficient	,286**	,416**	,397**	,556**	1,000	,663**	,440**	,502**
		Sig. (2-tailed)	,005	,000	,001	,000	.	,000	,000	,000
		N	93	143	71	261	377	104	275	183
	Calendar	Correlation Coefficient	.	,502**	,379**	,344**	,663**	1,000	,316**	,616**
		Sig. (2-tailed)	.	,000	,006	,001	,000	.	,003	,000
		N	27	74	52	93	104	120	88	69
	Location	Correlation Coefficient	,531**	,501**	,135	,455**	,440**	,316**	1,000	,317**
		Sig. (2-tailed)	,000	,000	,300	,000	,000	,003	.	,000
		N	98	153	61	295	275	88	457	164
	SMS	Correlation Coefficient	,273	,350**	,472**	,419**	,502**	,616**	,317**	1,000
		Sig. (2-tailed)	,064	,000	,001	,000	,000	,000	,000	.
		N	47	114	48	157	183	69	164	196
	Phone	Correlation Coefficient	,049	,564**	,451**	,506**	,641**	,443**	,596**	,713**
		Sig. (2-tailed)	,723	,000	,000	,000	,000	,001	,000	,000
		N	55	92	59	122	158	57	140	119
	Photos	Correlation Coefficient	,443**	,552**	,484**	,467**	,470**	,409**	,494**	,423**
		Sig. (2-tailed)	,000	,000	,000	,000	,000	,000	,000	,000
		N	159	175	59	369	301	96	345	167
	Camera	Correlation Coefficient	,373**	,438**	,475**	,413**	,466**	,321**	,431**	,337**
		Sig. (2-tailed)	,001	,000	,000	,000	,000	,008	,000	,000
		N	77	136	50	255	245	67	291	153
	Microphone	Correlation Coefficient	,163	,498**	,543**	,595**	,491**	,317*	,517**	,409**
		Sig. (2-tailed)	,231	,000	,000	,000	,000	,015	,000	,000
		N	56	113	45	193	182	58	196	140
	Wifi	Correlation Coefficient	,484**	,334**	,259*	,401**	,289**	,417**	,300**	,311**
		Sig. (2-tailed)	,000	,000	,016	,000	,000	,000	,000	,000
		N	142	182	86	305	244	79	276	159
	Bluetooth	Correlation Coefficient	,375**	,284**	.	,354**	,103	.	,346**	,191
		Sig. (2-tailed)	,011	,020	.	,001	,397	.	,002	,180
		N	45	67	55	91	70	41	79	51
	ID	Correlation Coefficient	,346**	,530**	,549**	,605**	,489**	,660**	,412**	,540**
		Sig. (2-tailed)	,001	,000	,000	,000	,000	,000	,000	,000
		N	94	146	55	232	197	58	212	152
	Other	Correlation Coefficient	,428**	,490**	-,099	,408**	,371**	,433**	,355**	,400**
		Sig. (2-tailed)	,000	,000	,548	,000	,000	,000	,000	,000
		N	174	160	39	361	292	77	351	152

**. Correlation is significant at the 0.01 level (2-tailed).
*. Correlation is significant at the 0.05 level (2-tailed).

FIGURE 5.5: Correlations between the app permission settings.

			Phone	Photos	Camera	Microphone	Wifi	Bluetooth	ID	Other
Spearman's rho	Purchase	Correlation Coefficient	,049	,443**	,373**	,163	,484**	,375**	,346**	,428**
		Sig. (2-tailed)	,723	,000	,001	,231	,000	,011	,001	,000
		N	55	159	77	56	142	45	94	174
	History	Correlation Coefficient	,564**	,552**	,438**	,498**	,334**	,284*	,530**	,490**
		Sig. (2-tailed)	,000	,000	,000	,000	,000	,020	,000	,000
		N	92	175	136	113	182	67	146	160
	Cellular	Correlation Coefficient	,451**	,484**	,475**	,543**	,259*	.	,549**	-,099
		Sig. (2-tailed)	,000	,000	,000	,000	,016	.	,000	,548
		N	59	59	50	45	86	55	55	39
	Identity	Correlation Coefficient	,506**	,467**	,413**	,595**	,401**	,354**	,605**	,408**
		Sig. (2-tailed)	,000	,000	,000	,000	,000	,001	,000	,000
		N	122	369	255	193	305	91	232	361
	Contacts	Correlation Coefficient	,641**	,470**	,466**	,491**	,289**	,103	,489**	,371**
		Sig. (2-tailed)	,000	,000	,000	,000	,000	,397	,000	,000
		N	158	301	245	182	244	70	197	292
	Calendar	Correlation Coefficient	,443**	,409**	,321**	,317*	,417**	.	,660**	,433**
		Sig. (2-tailed)	,001	,000	,008	,015	,000	.	,000	,000
		N	57	96	67	58	79	41	68	77
	Location	Correlation Coefficient	,596**	,494**	,431**	,517**	,300**	,346**	,412**	,355**
		Sig. (2-tailed)	,000	,000	,000	,000	,000	,002	,000	,000
		N	140	345	291	196	276	79	212	351
	SMS	Correlation Coefficient	,713**	,423**	,337**	,409**	,311**	,191	,540**	,400**
		Sig. (2-tailed)	,000	,000	,000	,000	,000	,180	,000	,000
		N	119	167	153	140	159	51	152	152
	Phone	Correlation Coefficient	1,000	,543**	,421**	,304**	,217*	,357**	,598**	,378**
		Sig. (2-tailed)	.	,000	,000	,002	,011	,006	,000	,000
		N	196	147	116	102	136	57	122	143
	Photos	Correlation Coefficient	,543**	1,000	,690**	,600**	,404**	,446**	,588**	,393**
		Sig. (2-tailed)	,000	.	,000	,000	,000	,000	,000	,000
		N	147	595	319	216	357	88	254	511
	Camera	Correlation Coefficient	,421**	,690**	1,000	,703**	,455**	,167	,363**	,478**
		Sig. (2-tailed)	,000	,000	.	,000	,000	,177	,000	,000
		N	116	319	370	203	230	67	181	299
	Microphone	Correlation Coefficient	,304**	,600**	,703**	1,000	,365**	,687**	,435**	,412**
		Sig. (2-tailed)	,002	,000	,000	.	,000	,000	,000	,000
		N	102	216	203	255	169	57	159	208
	Wifi	Correlation Coefficient	,217*	,404**	,455**	,365**	1,000	,642**	,382**	,510**
		Sig. (2-tailed)	,011	,000	,000	,000	.	,000	,000	,000
		N	136	357	230	169	462	105	262	353
	Bluetooth	Correlation Coefficient	,357**	,446**	,167	,687**	,642**	1,000	,245*	,369**
		Sig. (2-tailed)	,006	,000	,177	,000	,000	.	,025	,001
		N	57	88	67	57	105	119	83	73
	ID	Correlation Coefficient	,598**	,588**	,363**	,435**	,382**	,245*	1,000	,497**
		Sig. (2-tailed)	,000	,000	,000	,000	,000	,025	.	,000
		N	122	254	181	159	262	83	306	258
	Other	Correlation Coefficient	,378**	,393**	,478**	,412**	,510**	,369**	,497**	1,000
		Sig. (2-tailed)	,000	,000	,000	,000	,000	,001	,000	.
		N	143	511	299	208	353	73	258	666

**. Correlation is significant at the 0.01 level (2-tailed).
*. Correlation is significant at the 0.05 level (2-tailed).

FIGURE 5.6: Correlations between the app permission settings continued.

this kind of task, which have also been used in related research, include for example SVM algorithms like *support vector classification* [208]. We tried out several classifications, and achieved the best results with a KNeighbors implementation using two neighbours according to the two states (allow/deny) the permissions can have. For each permission, we trained a separate classifier, which was compiled into one *distributional machine learning* algorithm that always selects the correct classifier based on the input variables (here based on the permission to be predicted), as described in the last chapter.

As we have seen in the last chapter, both the personality measures as well as the IUIPC privacy measures are potential candidates as an input for machine learning. Having five personality measures and three privacy measures, this leads to a total of $2^8 = 256$ possible combinations of input features. Unlike the social network study, where we had a notably larger number of combinations, we are not dependent on a selection heuristic like the WSS algorithm presented in the social media study (see Chapter 4). We therefore tried out all possible combinations of input factors using a brute force method.

5.4.1 Comparative evaluation of the a priori permission prediction

We used a ten-fold cross-validation to train and validate the machine learning algorithm. In this method, the data set is first divided into ten parts of a similar size, called the *folds*. The training and evaluation phase is then performed ten times. In each run, one of the folds (later called the *validation fold*) is taken out for the validation at the end of the run. The other nine folds (later called *training folds*) are used for training and tuning the algorithm and selecting the optimal input features. To be more precise, we used 80% of the training folds for the training of the algorithm, and 20% for selecting the optimal set of input features. After the training, the trained algorithm has to predict the permission settings based on the input values of

Feature set	Selected features
IUIPC	Collection, Control
Personality	Extraversion, OpenExperiences

TABLE 5.3: Features selected by the machine learning algorithm.

the validation fold (in our case the individual features). The computed settings and the actual settings from the validation fold are then compared to compute the precision of that fold. At the end, the precision of the ten folds is again averaged to gather the final precision of the prediction. The input variables selected by the algorithm (see Table 5.3) correspond to the individual measures with the highest correlation, which supports the correctness of the input selection method and the results of the correlation analysis.

To get an impression of the quality of the prediction, we conducted a comparative evaluation, where we compared the precision of the a priori permission prediction, using either the personality or the IUIPC privacy measures, with a naive approach suitable for this scenario that we call the *random probabilistic method*. In a simple random approach, 50% of the settings would be set to "allow", and the other 50% to "deny" on average. However, according to the results in Table 5.2 the percentage of denied and allowed permissions is not about 50% on average, so a naive random approach would lead to an unfair comparison. For the *probabilistic random approach*, we perform the same steps as for the machine learning approach, including a training phase and a ten-fold cross-validation. In each run of the cross-validation, we use the nine training folds to compute the average percentage of denied permissions for each permission type. In the validation phase, the *random probabilistic approach* then decides to allow or deny the permission based on the percentage calculated in the training phase, i.e. when the permission is denied in 80% of the cases in the training set, the approach will decide to deny the permission with 80% probability, and allow with a probability of 20%.

5.4.2 Results

The results of the comparative evulation are shown in Table 5.4. The columns denote the different approaches, namely the *random probabilistic approach* and the personalized machine-learning-based prediction using the *IUIPC* privacy measures or the *big five personality measures*. The rows denote the different permission types that were predicted, in addition to an *all* row denoting the average precision for all permissions. The values shown are the percentages of correct predictions.

Although the *probabilistic random approach* performs notably better ($M = 59,64$) than a naive random approach (whose accuracy would be about 50%), this approach is outperformed by the personalized machine-learning approach by more than 10%, using either the personality or privacy measures ($M_{IUIPC} = 70.92$, $M_{Personality} = 69.37$). The permissions that could be predicted with highest accuracy are the *Bluetooth* ($M_{IUIPC} = 96.66$, $M_{Personality} = 93.33$) and *Cellular Information* permissions ($M_{IUIPC} = 92.5$, $M_{Personality} = 91.25$). The most errors were made predicting the *location* permission ($M_{IUIPC} = 53.33$, $M_{Personality} = 58.48$), as well as access to SMS ($M_{IUIPC} = 50$, $M_{Personality} = 57.5$) and the phone functionality ($M_{IUIPC} = 67.33$, $M_{Personality} = 58.66$) of the smartphone. Interestingly, some of the permissions, mostly the ones that are perceived as sensitive (i.e. that have a high denial rate according to Table 5.2), can be predicted better using the personality measures, for

Permission	Random	IUIPC	Personality
All	59.64	70.92	69.37
Purchase	59.37	78.13	67.50
History	65.88	72.94	78.82
Cellular	78.75	92.50	91.25
Identity	51.87	68.44	60.62
Contacts	48.88	55.18	64.44
Calendar	70.00	80.00	81.11
Location	45.15	53.33	58.48
SMS	54.37	50.00	57.50
Phone	53.33	67.33	58.66
Photos	47.31	63.65	62.44
Camera	53.92	60.00	61.07
Microphone	52.50	74.00	69.00
Wifi	68.82	86.47	78.82
Bluetooth	84.44	96.66	93.33
ID	56.08	64.78	58.70
Other	63.55	71.33	68.22

TABLE 5.4: Prediction accuracy (percentage of predictions that are correct) for prediction with the Random Probabilistic Model (Random), and prediction using the IUIPC questionnaire or the big five personality test.

example the SMS and Contact ($M_{IUIPC} = 55.18$, $M_{Personality} = 64.44$) permissions, whereas other permissions like Phone ($M_{IUIPC} = 67.33$, $M_{Personality} = 58.66$) and Wifi ($M_{IUIPC} = 86.47$, $M_{Personality} = 78.82$) can be predicted better using the IUIPC privacy measures.

5.5 Dynamic setting prediction

Whereas the a priori permission prediction discussed in the last section targets the first use-case, where a new user wants to set up *all permissions* for a newly bought smartphone, *dynamic setting prediction* targets the second use-case, where a user has already set up most of his privacy settings, and wants to adapt the privacy settings of just one newly installed app. For this purpose, we are facilitating pairwise correlations of privacy choices between each combination of two permissions, so that the choice of one or more permission can be used to predict the choice of the other. In Android OS or iOS, the permissions of an app are typically displayed as a scrollable list, where all permissions are enabled by default and listed one after another. The user then traverses the list and clicks on a permission he wants to deny. In this case, our approach takes all permission choices up to this point, and uses them as an input to predict the choices for the remaining permissions. Figure 5.7 illustrates the functionality of this approach with an example using the Evernote app. The user decides to deny the *Contacts* permission. Based on this input, the dynamic setting prediction assumes that the user intentionally left all other permissions above "Contacts" enabled, so it takes the calendar, camera, and contacts permissions as an input to predict the choice of the remaining permissions below the "Contacts" permission. This approach will later be called *dynamic setting prediction*. Technically, we implemented the approach similarly to the *distributional machine learning* technique

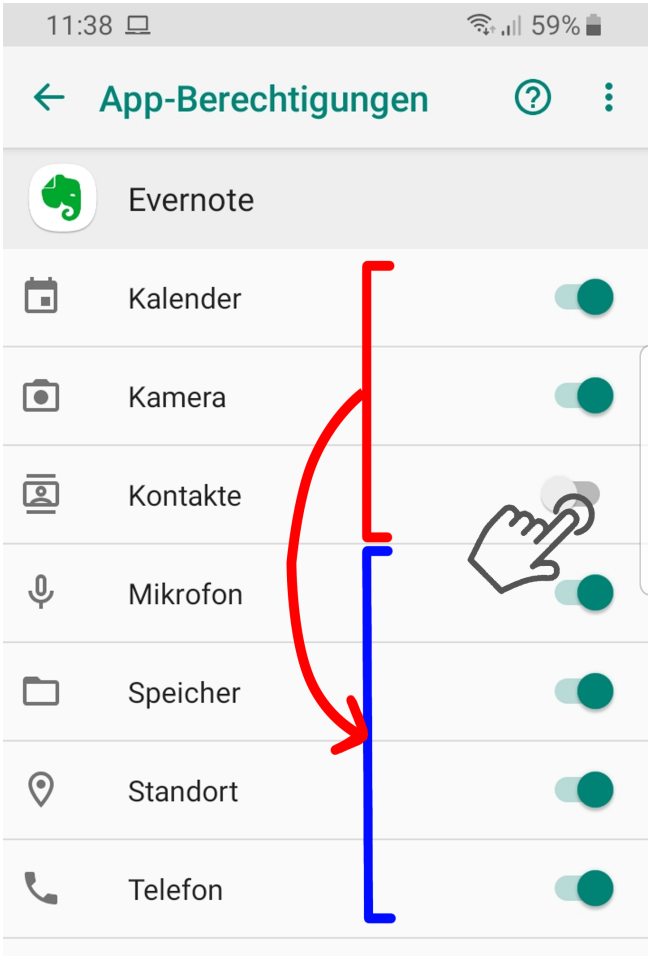


FIGURE 5.7: Functionality of the dynamic settings prediction for the example of the Evernote app. The user traverses the list of permissions starting from the top and adapts the "Contacts" permission setting. The dynamic settings prediction uses the permissions already checked by the user (red) as an input to predict the remaining unchecked settings (blue).

described in Chapter 4: For each combination of input permissions, we trained a separate predictor using the training data. Whenever the user changes a permission and a prediction is requested, the algorithm loads the machine learning estimator according to the given input permissions, and returns the prediction result of this estimator.

5.5.1 Evaluation of the dynamic setting prediction

We validated the approach using our study data by simulating a user's behavior when choosing the permission settings, and comparing the number of *clicks* needed either *with the dynamic setting prediction*, or *without* additional support. As a *click*, we count every user interaction, e.g. every permission setting that he had to change manually. For the unsupported version, the number of clicks therefore corresponds to the number of denied permissions (i.e. the ones which the user had to change from *allow* to *deny*). A simplified version of the program used for evaluating the *dynamic setting prediction* is described below.

```
# traverse all the user settings
for each user_settg in testset:

 # initially, all settings are
 # set to "allow"
 pred=allow_all

 for each perm in user_settg:
  if user_settg[perm]!=pred[perm]:

   # prediction was wrong,
   # user had to change the setting
   # -> predict remaining settings

   pred[perm]=user_settg[perm]
   predict_settings_below()
```

The test set contains all correct app settings from the user study, and is loaded by the evaluation program on startup. The program then traverses each of those user settings (user_settg). Initally, all permissions are set to *allow*. Then, all permissions for the app are traversed. As soon as one of the permissions is denied, a click is recorded and the *distributional machine learning algorithm* is fed with the permissions up to this point as an input, and has to predict the remaining permissions. The validation program then continues traversing the permissions. If one of the remaining (predicted) permission settings differs from correct user_settg, a click is recorded (as the user would have to adapt the incorrectly predicted setting), and the remaining permissions are again predicted using the already traversed and checked permissions. The algorithm stops if the last permission from the last app (i.e. the last user_settg) is traversed.

In the study, we compared the average number of clicks needed without support and with the *dynamic setting prediction* using either only the permissions, the IUIPC privacy measures, the personality measures, or all of them together as an input for the prediction. For each comparison, we report the average number of clicks needed either with or without support, and in how many cases the dynamic setting prediction needed fewer clicks (*Won %*), the same amount (*Draw %*) or more clicks (*Lost %*) than the unsupported counterpart. As the *dynamic setting prediction* acts only if at least one of the permissions is changed, we used only app settings for the evaluation where at least one permission is changed. For the remaining cases, the number of clicks would be the same either with or without support.

5.5.2 Results

Table 5.5 shows the results of the comparison. Regardless of the input variables used, the *dynamic setting prediction* always performed better than the unsupported method, both regarding the percentage of comparisons won against comparisons lost, as well as the average number of clicks. However, the precision differs between the input methods, especially the percentage of lost comparisons. Using only the permissions for the prediction, the *dynamic setting prediction* lost in 17 % of the cases. Using either the personality or privacy measures leads to a loss rate of about 14 %. If all features are used together, the dynamic setting prediction loses in only 8 % of all cases. However, in about 60 % of all cases, the number of clicks is similar to

Input	Won %	Draw %	Lost %	Clicks (supp.)	Clicks (unsupp.)
Only Permissions	23.49	59.40	17.15	1.91	2.22
IUIPC	25.76	60.60	13.63	1.83	2.21
Personality	26.58	59.30	14.12	1.70	2.10
All	24.66	67.23	8.11	1.58	2.00

TABLE 5.5: Results of the dynamic settings prediction with the different sets of input features.

that using the unsupported method. Similar to the loss rate, the number of clicks also drops from 1.91 using only the permissions to 1.83 and 1.70 using the privacy and personality measures, respectively, and can be reduced to 1.58 clicks on average using the dynamic setting prediction compared to 2 clicks without any support.

5.6 Discussion and limitations

The study results show that there are significant correlations between the chosen permission settings and the personality measures as well as the privacy measures. According to the machine learning analysis, those correlations are strong enough to perform a prediction of the privacy settings that is notably more precise than a random prediction, and also better than the *random probabilistic approach*, taking the permission type and its typical denial rate into account. To be more precise, we were able to achieve an accuracy of up to 71% correct predictions using only the three privacy measures. However, there are still limitations to the approach and the user study, which we will discuss in this section.

5.6.1 Implementation of the proposed approaches

We presented and evaluated *two* approaches to support the user in doing his mobile phone permissions settings: The first is an a priori permission prediction that helps the user in choosing *all* privacy settings at once, and the second assists the user while actively adapting the privacy settings for a specific app. In a final implementation of the a priori permission prediction, the user could start by filling out the twelve questions of the IUIPC privacy questionnaire, which could then be used for calculating the three IUIPC privacy measures, that will be used by the prediction algorithm later. Alternatively, the user could grant access to his Facebook profile or a blog website, which could be used for deriving the personality or the privacy measures (see chapter 3). The prediction algorithm then computes the permissions in the background, and displays the results in an overview UI, where the user can review and polish the settings before applying them. Both lay users as well as technically adept users can benefit from this system. Lay users profit from the fact that the a priori permission prediction does not pose any technical questions (like "with which kind of app would you like to share you cellular information?"), so they do not need the technical knowledge on what the permission means, and which effect the denial or allowance of the permission might have on the app, their phone security and their data privacy. Both lay and expert users profit from the reduced amount of work that has to be done, as only twelve IUIPC questions have to be answered instead of adapting almost 500 distinct permission settings for every app. The second approach, called *dynamic setting prediction*, could be integrated into the permission dialogue of the Android OS. Whenever a user opens the permission settings page

of an app, traverses the list and changes a setting, the dynamic setting prediction is activated, and checks which of the remaining settings should also be denied. The changed settings are then marked in orange, so that the user can see and review the settings that have been changed by the algorithm. According to the study results, this improvement of the permission dialogue should save smartphone users many clicks, and hence reduce the time needed to adapt the privacy settings of a newly installed app, and therefore also decrease the experienced frustration and increase the motivation in adjusting the privacy settings, as it takes less time.

5.6.2 Both questionnaires should be used

The personality as well as the privacy questionnaire both achieve good results when used as an input for the permission prediction. The privacy measures even perform slightly better than the personality measures in our study. Nevertheless, as the difference between the questionnaires is very small and not significant, we cannot clearly state that the privacy measures should be used for this kind of prediction. We would still like to find out whether there is a significant difference in precision between the questionnaires in future work. Furthermore, taking a closer look at the prediction accuracy for the different permissions, one can see that some of the permissions can be predicted better using the IUIPC questionnaire, whereas others (like Contacts, Locations, SMS) are predicted better using the privacy measures. We therefore speculate that the "best" questionnaire depends on the permission that has to be predicted, so that a final solution should always record both the privacy and personality measures in combination to achieve best results. According to our results, the prediction precision can be increased to 72.8% if the best questionnaire is selected for each permission individually. Whether this claim holds has yet to be proven in future work.

5.6.3 Size of the training set and combination with other approaches

Most of the approaches in related work and also in other domains that are targeting the cold start problem, e.g. the need to predict privacy settings when no data on past privacy decisions is available as a source for the prediction, mainly concentrated on context factors for the prediction, and did not offer a personalized prediction as in our case. Such context factors are not individual, and can therefore be easily collected for example by using an app that is allowed to send the chosen permissions to researchers in an anonymized form. This is not possible for individual measures, like personality or privacy measures, that have to be captured manually from each user using questionnaires. Studies of that kind, for example the one conducted by Bin Liu et al. [208, 206] therefore had access to a large database with millions of app permissions and the respective context factors (for example the app category or the purpose of the permission, according to a static code analysis) and were therefore more precise than the personalized approach presented in this chapter.

The goal of our study was *not* to test how good an individualized prediction can be using a large database; the goal was to test *whether* individual factors are a useful input feature for such a prediction. As the power analysis of the correlation analysis has shown, about 1400 participants would be needed to achieve high statistical power for *all* correlations. Our prediction did not make use of any of the context factors; still, we were able to increase the prediction by about 20% compared to a purely random method. We therefore suggest using the approaches from related literature based on context factors, and improving them by also introducing individual factors

as a source of prediction, so that on the one hand, the precision can be increased in general, and it is also tailored to the user's personality and individual privacy desires on the other hand. We would like to test how the precision increases with a large data set, and especially which precision rates can be achieved when combining both context-based and individualized factors for the prediction, in future work.

5.6.4 Control of random variables

Notably, we took care to reduce the number of random variables to a minimum. For the online, study, we used prescreening techniques to make sure that the participants were using a stationary computer or a laptop and that they were at home while filling out the questionnaire, so that disturbances from their surroundings were minimized. Still, there are other random variables that we cannot control, like a general distrust towards specific app developers or types of apps, that differ between users. By capturing personality measures within the study, we were also able to check whether the personalities of the participants were different from a normal distribution. We compared the recorded personality measures with personality measures from earlier studies, and found no significant difference; we therefore assume that we do not have any personality biasing in our test set.

5.6.5 Denied permissions per app and precision of the dynamic setting prediction

Taking a look at the study results, one can see that the dynamic settings prediction rarely loses against the unsupported approach, but often has the *same* number of clicks. Having a look at the amount of denied permissions per app in Figure 5.2, one can see that from the cases where at least one permission was denied, only one permission has been denied in 33% of all cases. The dynamic setting prediction comes into action only after the first permission is denied; therefore the algorithm cannot win in this situation: Either the prediction is correct (i.e. it does not propose to deny any other permission), and then the amount of clicks is the same as for the unsupported method; or the prediction incorrectly proposes to deny a further permission, leading to an additional click for the user to change the setting back to allow. Another 20% of the permissions have only two denied permissions, which is also difficult for the dynamic setting prediction. Although the conditions were therefore difficult for our approach, it was still equally good or better in up to 92% of all cases when using all input features.

5.6.6 Precision of the prediction depends on the permission

The study results in Table 5.4 show that the precision of the prediction differs greatly between the permissions. That holds for the machinellearning-based prediction, as well as for the random probabilistic approach. In the cases where the number of denied permissions is high, for example for the *Location*, *SMS* and *Contacts* permissions, it is very hard for the probabilistic model to achieve good results, whereas it is easy for permissions which are rarely denied. Particularly hard cases, where many users struggle with allowing access to their sensitive data, are the cases where the personalized approach, based on machine learning, can outperform a naive unpersonalized solution like the random probabilistic approach most. The results therefore indicate that users profit from our approach especially for sensitive permissions, where incorrect settings can lead to the greatest damage to the user's privacy.

5.7 Conclusion

Modern smartphone operating systems allow every single permission to be set for each app, which on one hand gives a lot of power to the user, but on the other hand also demands too much from her, especially if she is a lay user that first has to understand the meaning and consequences of the permissions. The case is especially hard if data about the user's privacy behavior, like permission settings from other apps or from online behavior, is not available. Other approaches have so far concentrated on context factors like the app category and permission type to infer permission settings in that case. However, although those approaches reached a good prediction precision, the prediction results were generic and not tailored towards the user's personality and privacy desires. We examined whether such individual factors, in our case the big five personality traits and the IUIPC privacy measures, can be used to derive personalized app permission settings for users. We performed a user study including 100 participants to gather the desired permission settings as well as individual measures, and described and evaluated *two* approaches to support the user during her permission setting process. The first is an a priori permission prediction that uses individual measures to predict *all* permission settings for all of the user's apps at once, which is helpful when she has just bought a new phone and has to adjust the settings. The second approach is a *dynamic setting prediction*, that actively supports the user when she has installed a new app and has to adapt the permission settings only for this app, by guessing future permission settings based on the ones chosen so far for the app.

The study results have shown that the a priori permission prediction performs about 10% better than a random probabilistic method that takes only the permission type and its typical denial rate into account. Especially for sensitive permissions, the a priori permission prediction is notably more precise than the unpersonalized approach. Also, the *random probabilistic approach* is better or equally good in 92% of all tested cases, if all individual features are used. Still, there is a lot potential for future work, from integrating other context factors into the approach, to deploying and testing the idea in an in-the-wild study, to observing whether spatial or temporal permission customization is required by users and can further increase the prediction precision.

Chapter 6

Predicting privacy settings using individual and context factors – in the intelligent retail domain

Since the beginning of retail business, retail data privacy has always been a big issue. Although convential retail stores already collect a lot of information like products purchased, number of customers, sales amounts and much more, the customer had the possibility to stay anonymous by paying in cash and not using any credit or loyalty cards. The retail companies rarely had a chance to match the sales data to the individual customers. Online shopping platforms, on the other hand, have the ability to match viewed, bought, sold and returned items to individual customer accounts. The recent launch of Amazon's first brick-and mortar retail store, called "Amazon Go", posed new privacy challenges due to the increased amount of private data recorded inside the store. Where the customer was able to stay anonymous during the shopping process in conventional stores, he is now tracked throughout the complete shopping journey: Upon entering the shop, the shopper uses the NFC functionality of his smartphone to identify himself at the entrance gate. He can then browse the store, grab products, put them back again, and just leave the store without going through a checkout process or scanning the products; he can just leave. Amazon achieves this using "sensor fusion and deep learning"[1] without providing further details. The technology behind the service is most likely based on camera systems and other sensors that follow the route of the customer from the entrance, where she is identified using her smartphone, along the different aisles and shelves that are visited, and where the system registers grabbed and viewed products, up to the exit where the system recognizes the contents of the shopping cart and automatically withdraws the price of the shopping cart items from the customer's credit card. Although these techniques saves a lot of time for the customers and allows the system to support them in finding the products they need, not all customers are happy with this system: In order to make the service work, Amazon has to record and store a large amount of private data throughout the shopping process, and it is not even anonymized. Which data exactly is recorded, where the data is stored and for what purpose Amazon uses it, is as unclear as the technology that is used for recording the data.

Apart from operating intelligent retail stores like Amazon Go, there exist several research laboratories, including the Innovative Retail Laboratory (IRL) [308], which investigate the capabilities of new technologies in the context of brick-and-mortar

[1] http://www.self.com/story/amazon-go-grocery-store-of-the-future
(last accessed: 2020-03-09)

retail stores. The Innovative Retail Laboratory (IRL) is an application-oriented research laboratory of the German Research Center for Artificial Intelligence (DFKI) run in collaboration with the German retailer GLOBUS SB-Warenhaus Holding in St. Wendel. In this living lab, research in a wide range of different domains is conducted, mostly related to intelligent shopping assistance. The demonstrators range from an instrumented shopping cart employing indoor navigation to several intelligent shopping consultants, ambient information services and an automated checkout system.

We have already seen, in the last chapters, that the usage of machine learning approaches can help to predict personalized privacy settings for a user, based on her personality and privacy measures. In this chapter, we try to tackle the problem of data privacy in intelligent retail stores by reducing the amount of needed input. In detail, we try to facilitate correlations between the personality or privacy desires of a customer and his desired retail privacy settings, in order to form a privacy assistant that is able to predict the appropriate individual privacy settings for a customer.

Unfortunately, the domain of intelligent retail data has not been a topic of research in the usable privacy domain before. To be able to make a prediction of the privacy settings, we therefore first had to find out *which* data is typically recorded inside an intelligent retail store, and which aspects are perceived as sensitive by the users.

In detail, we try to solve the following research questions in this section:

1. Which data is collected in current or yet-to-be-built intelligent retail stores?
2. Is there a correlation between personality or privacy attitudes and customers' data disclosure preferences for an intelligent retail store?
3. Can the correlation be used to predict the data disclosure preferences using a short personality questionnaire like the IUIPC or TIPI?
4. Are customers interested in having control over their own recorded data in an intelligent retail store, or do they trust in retail companies?
5. Do customers accept a privacy UI which helps them to monitor and tune their privacy settings for the disclosure of their private data in intelligent retail stores?

In the last chapters, we have already seen that in some domains, like social networks, location sharing or mobile app permissions, there is a correlation between a user's personality and privacy measures and their choice of privacy settings. In this chapter, we examine whether this concept can also be transferred to the domain of intelligent retail data.

In a first step, we used expert information from members of the Innovative Retail Lab to create a list of privacy-sensitive data items that is used inside an intelligent retail store. We then conducted a larger user study including 100 participants to first check for correlations, and then to train a machine learning component to make a prediction of these.

Although the main focus of our work is on the prediction of data disclosement preferences, we present a user interface which helps the user to set his privacy settings for retail data in a centralized system. Machine learning is utilized to help the user to find his optimal settings in a privacy assistant, by taking the user's personality as a basis for the prediction. The results of the user interface evaluation show that

the UI including machine learning support is perceived as more comfortable and is significantly preferred to a standard UI without machine learning.

For this purpose, we conducted a user study consisting of two different stages: First we had to gather background knowledge about data usage and privacy issues in intelligent retail stores. Afterwards, we conducted an online user study to find correlations between the personality or privacy attitudes of a user and the privacy settings for the aforementioned data. Privacy measures were captured using the IUIPC questionnaire; personality was determined using the big five personality measure [78] in the form of the Ten Item Personality score (TIPI) [137] questionnaire, which is a shorter version of the original big five personal inventory questionnaire (NEO-PI-R). Although the possibility exists to extract the personality measures out of written text, we decided to capture them using a questionnaire in our study to reduce side effects. Apart from these questionnaires, we posed two additional questions about privacy and privacy invasion (see Table 6.3). To be more precise, we asked the subjects how frequently they had been a target of a privacy invasion (on a five point ordinal scale from very frequently to never), and how often they enter wrong information on purpose on websites (percentage as a numeric scale). The two stages of the study are described in the next two subsections. The work presented here is based on already-published research [268].

6.1 Background analysis: Data items recorded inside an intelligent retail store

Prior to the main user study, we reviewed the exhibitions of the Innovative Retail Laboratory [308], to find out which data is gathered inside the IRL and could be recorded in other intelligent retail stores, and created a list of privacy-sensitive data items, later called *permissions* or *items* within the *retail privacy settings*. The data items found in the review are shown in Table 6.1 together with a short description, whereas Table 6.2 shows a list of services that are present in the IRL, as well as the data that is recorded or required for the service to work.

Most data is recorded for the "invisible checkout", which allows the customer to just grab products out of the shelves, and to leave the store without the need to scan and pay for the products at a checkout. The IRL relies on several methods like RFID tags inside the products or optical sensors and cameras, for example, to find out which products have been placed inside the shopping cart. In addition to the viewed and bought products, the IRL also keeps track of the shoppers' route inside the store, including visited areas and stopping points. The IRL uses a Bluetooth location system called Quuppa[2] for this purpose. The data allows generating heatmaps for a "management dashboard", which allows the store manager to optimize the store layout, for example. The Innovative Retail Lab offers several recommender systems to the customer, which recommend products that fit with the other products inside the shopping basket, or that match the client's typical product set. In addition, it is possible to highlight allergy information on the products inside the store. For this purpose, nutrition preferences and allergy information about the customer are stored.

[2] http://quuppa.com/ (last accessed: 2020-03-09)

Variable	**Description**
Address	Personal information of the customer
Birthday	
Name	
(Household) income	
Nutrition	Nutrition/product preferences like vegan/vegetarian, likes fish, dislikes meat
Allergies	Customer's allergies
Recent visits	Date, time and place of the last shop visits of the customer
Wishlist	Bookmarked items/items on the customer's shopping list
Recently viewed	Items that have been recently viewed by the customer, e.g. taken from the shelf and put back
Receipt	Detailed shopping receipt, including the products bought with their exact names and product IDs
Category	The categories of the products bought, e.g. "vegetables" or "cereals"
Amount	The amount of products bought
Price	The price of each of the products bought
Loyalty	Loyalty points
Location	In-store location and movement pattern of the customer

TABLE 6.1: Private data that is recorded in an intelligent retail store.

Service	Data
"Invisible" Checkout	- Address - Birthday - Name - Recent visits - Recently viewed - Receipt - Category - Amount - Price - Loyalty
Digital shopping list	- Wishlist
Customer heatmap/ customer flow for market manager	- Location
Allergy advisor	- Allergies
Product recommender	- Nutrition - Income

TABLE 6.2: IRL services and private data used.

Label	Question
Falsify	Some websites ask you for personal information. When asked for such information, what percent of the time would you falsify the information?
Invasion	Have you ever been the target of a privacy invasion (e.g. your data was misused or shared without your knowledge)?

TABLE 6.3: Question text and label of the additional question set.

6.2 Pilot study

The pilot study was conducted with five participants recruited from the university context. All of them were students aged between 21 and 48 (average 38). Like the IRL review, the study has an explorative nature, and has to be seen as a qualitative study to get a first impression on possible *data groups* and sensitivity orders, without a claim of absolute correctness. The results will be validated within the interface evaluation study later. The study was done using a questionnaire, which was constructed as follows: In the first question, the participants were given the list of retail data types along with a set of category names (app data, personal profile, location data, sales receipt data, interests). The participants then were asked to either assign the data types to a group, or to create a new group. As our list might not be exhaustive, we asked whether there were other types of data that might be recorded that came to a participant's mind, and which data types would be hard to assign to a specific group. The next question asked about the sensitivity of the different data types on a five-point scale from "I would never disclose this data" to "I would disclose this data without any concerns".

All proposed clusters were used, except for the "app data" cluster, which was perceived as too vague by most of the participants. They agreed to assign the "app data" to the data group according to the *type* of data, e.g. whether it is location data or related to the sales receipt. Apart from that, they had no problems assigning the items to the proposed group, and did not feel the need to create new groups. The clusters and sensitivity ratings for the different data types are shown in Table 6.4.

Except for personal data, we were able to bring the data types inside a cluster into an ascending order regarding the reported sensitivity. The sensitivity for personal data was too varied to find a meaningful order that works for all participants. In the user interface presented in Section 8, we showed the data items with descending sensitivity, so that the users' focus lies on the most sensitive data items and so that they are checked and adapted first.

6.3 Online study

Based on the results of the expert interview, we were able to design an online study to check for correlations between privacy attitude and data disclosement behavior. The study was conducted as an online survey using the software LimeSurvey[3]. 100 participants were recruited using Prolific Academic[4]. each participant received

[3] https://www.limesurvey.org, last accessed 09-05-2016
[4] https://www.prolific.ac/, last accessed 09-05-2016

		P1	P2	P3	P4	P5	Rank
Personal Data	*Address*	3	4	3	4	4	
	Birthday	3	1	2	3	2	
	Name	3	2	2	4	4	
	Income	5	3	3	4	3	
	Gender	3	1	2	3	2	
	Education	3	3	3	4	4	
Location data	*Recent visits*						
	- Province	1	1	2	3	3	1
	- City	2	2	2	3	3	2
	- Address	3	2	3	4	3	3
	Movement	4	1	2	3	3	4
Shopping Receipt	*Loyalty points*	1	2	3	3	2	1
	Items bought						
	- Amount	4	3	3	2	3	2
	- Category	3	3	3	3	2	3
	- Price	3	2	3	3	2	4
Interests	*Wishlist*	1	3	3	3	2	1
	Recently viewed	2	3	3	3	2	2

TABLE 6.4: Sensitivity rankings for the different data items from 1 (very sensitive) to 5 (very unsensitive) and sensitivity ranking inside each *data groups*. Data items are listed with descending sensitivity in the user interface to put the focus of the user on sensitive data items.

2£upon completing the questionnaire succesfully. The recruiting system from Prolific Academic allowed us to check the results for plausibility before the participant was paid. If we rejected the results, for example because one of the control questions was not answered correctly, the system automatically recruited a new participant. Therefore we have exactly 100 viable results. The age of the participants ranged from 18 to 73 years (average 33, SD 11.7). The audience had a wide variety of occupations: we recruited students, self-employed workers, employees, and also homemakers.

The survey contains two parts: In the first part, we asked the subjects to fill out the privacy and personality questionnaires. In the second phase, we asked, for each item of the *Retail privacy settings*, how likely he or she would refuse to disclose the item on a six-point scale (1 = very unlikely, 6=very likely). The survey ended with a short feedback question in free-text style.

6.4 Results

The 100 participants filled out 100 *retail privacy settings*, whose mean and standard deviation can be found in Table 6.5 together with the frequency of denied permissions (sharing likelihood < 3).

As our study data consists of ordinal variables and was not normally distributed according to F-tests, we decided to perform a Spearman correlation ("Spearman's Rho") on the individual measures and the privacy settings. The results are shown in Figure 6.1.

The measures of the privacy and personality questionnaires are in the rows, whereas the retail privacy settings are plotted as the columns of the table. Significant and highly significant correlations are marked with one or two asterisks, and colored

Item	mean	stdev	% denied
Name	3.96	1.48	29.7
Birthday	3.79	1.66	36.6
Address	3.00	1.71	60.4
Income	2.97	1.59	60.4
Nutrition	4.37	1.47	23.8
Allergies	3.74	1.70	40.6
Recent visits	3.68	1.49	39.6
Wishlist	3.78	1.62	41.6
Recently viewed	3.96	1.48	54.5
Receipt	3.79	1.66	31.7
Category	3.00	1.71	21.8
Price	2.97	1.59	24.8
Amount	4.35	1.33	22.8
Loyalty	4.37	1.47	15.8
Location	3.74	1.70	66.3

TABLE 6.5: Mean and standard deviation for the sharing likelihood and percentages of denials for each retail privacy setting.

in gray or dark gray, respectively. Regarding the IUIPC measures (collection, control, awareness), the collection measure yields highly significant correlations for most of the permissions (10 out of 16). Control and awareness both also correlate with several permissions. The general personality seems not to correlate with the choice of retail privacy settings and is therefore unsuitable for a machine-learning-based prediction. However, the amount of falsified information given to online companies seems to correlate significantly or highly significantly with seven out of 14 items of the privacy settings. We therefore dropped the TIPI questionnaire and continued to work with the IUIPC and our additional questionnaire for the prediction in the next sections.

6.5 Retail privacy setting prediction

As the results in the last section have shown, there is a correlation between individual measures and privacy settings, whose suitability for predicting the privacy settings using machine learnings will be checked within this section. Similar publications predicting privacy settings in other domains [208] used support vector algorithms for the prediction. As we have fine-grained ordinal data for the privacy settings, we are also able to use a regression to predict privacy settings. The prediction result is then mapped to allow (result $>$ 3.5) or deny (result $<$ 3.5). We tried several regression methods and achieved the best results with a ridge regression.

We followed the same procedure described in earlier section, i.e. training the algorithm, adjusting parameters and finally validating the trained regressor. To prevent biasing of the results, we used a cross-validation method called *repeated random sub-sampling validation*, also known as *Monte Carlo cross-validation*: The data set is split into two parts, the *training set* containing 75% of the data set, and the *test set*, that is used solely for the evaluation, containing 25% of the data. The data in the *test set* is never used while setting up the algorithm, neither for training/fitting, nor for selecting optimal algorithm parameters, nor for finding the optimal feature set. We

		Name	Birthday	Address	Income	Nutrition	Allergies	Recent _visits	Wishlist	Recently _viewed	Receipt	Category	Price	Amount	Loyalty	Location
Collection	Correlation Coefficient	-,325*	-,395*	-,396*	-,410*	-,204*	-,350*	-,387*	-,256*	-,298*	-,270*	-,100	-,141	-,074	-,104	-,478**
	Sig. (2-tailed)	,001	,000	,000	,000	,041	,000	,000	,010	,002	,006	,320	,160	,464	,301	,000
	N	101	101	101	101	101	101	101	101	101	101	101	101	101	101	101
Control	Correlation Coefficient	-,216*	-,271**	-,248*	-,302**	-,107	-,193	-,371**	-,102	-,325**	-,386**	-,059	-,030	-,001	,048	-,476**
	Sig. (2-tailed)	,030	,006	,012	,002	,289	,053	,000	,310	,001	,000	,555	,766	,990	,631	,000
	N	101	101	101	101	101	101	101	101	101	101	101	101	101	101	101
Awareness	Correlation Coefficient	-,191	-,249*	-,109	-,206*	-,109	-,162	-,250*	,004	-,238*	-,321**	-,032	,013	,027	,001	-,326**
	Sig. (2-tailed)	,056	,012	,278	,039	,280	,106	,012	,966	,016	,001	,750	,899	,791	,991	,001
	N	101	101	101	101	101	101	101	101	101	101	101	101	101	101	101
Extraversion	Correlation Coefficient	,144	-,028	,035	,157	,000	-,018	,081	,028	,076	,025	,100	,067	,127	-,007	,037
	Sig. (2-tailed)	,149	,780	,725	,116	1,000	,857	,418	,780	,450	,803	,322	,509	,207	,946	,717
	N	101	101	101	101	101	101	101	101	101	101	101	101	101	101	101
Agreeableness	Correlation Coefficient	-,078	,081	-,086	-,054	,007	,034	-,028	-,056	-,104	,001	,100	,046	,094	,138	-,131
	Sig. (2-tailed)	,435	,423	,392	,591	,943	,735	,777	,580	,299	,990	,318	,648	,349	,169	,190
	N	101	101	101	101	101	101	101	101	101	101	101	101	101	101	101
Conscientiousness	Correlation Coefficient	-,123	-,024	-,078	,032	-,052	-,061	-,004	-,045	-,151	-,145	-,010	-,071	-,039	,015	-,120
	Sig. (2-tailed)	,219	,812	,438	,749	,609	,542	,966	,653	,131	,148	,917	,479	,695	,882	,231
	N	101	101	101	101	101	101	101	101	101	101	101	101	101	101	101
Neuroticism	Correlation Coefficient	-,014	-,038	,013	-,051	-,089	-,026	-,055	,038	,025	,053	-,061	-,043	-,116	,053	-,069
	Sig. (2-tailed)	,889	,709	,898	,616	,376	,796	,585	,707	,801	,597	,545	,670	,248	,599	,492
	N	101	101	101	101	101	101	101	101	101	101	101	101	101	101	101
Openness	Correlation Coefficient	,010	-,042	-,022	,073	,043	-,002	-,089	,221*	,146	,124	,019	,242*	,185	,205*	-,131
	Sig. (2-tailed)	,922	,677	,825	,467	,670	,985	,376	,026	,146	,218	,852	,015	,064	,040	,191
	N	101	101	101	101	101	101	101	101	101	101	101	101	101	101	101
Invasion	Correlation Coefficient	-,032	,002	,036	-,045	-,017	-,011	,096	-,025	,044	-,019	,021	,014	-,129	-,013	,183
	Sig. (2-tailed)	,752	,982	,722	,653	,868	,913	,338	,805	,666	,848	,833	,887	,197	,898	,067
	N	101	101	101	101	101	101	101	101	101	101	101	101	101	101	101
Falsify	Correlation Coefficient	-,193	-,300**	-,275*	-,067	-,082	-,224*	-,172	-,168	-,213*	-,191	-,167	-,226*	-,199*	-,319**	-,108
	Sig. (2-tailed)	,053	,002	,005	,505	,415	,024	,085	,094	,032	,056	,095	,023	,046	,001	,282
	N	101	101	101	101	101	101	101	101	101	101	101	101	101	101	101

FIGURE 6.1: Correlations between privacy awareness/personality measures and retail privacy settings. Entries in light and dark gray are statistically significant and strongly significant, respectively.

performed 100 distinct runs, and used the average precision of all runs for selecting the best set of features. After each run, the data set was shuffled randomly, and reassigned to one of the two parts.

6.6 Validation

We validated the results of the prediction using the trained estimators as described in the last subsection. For this purpose, we used the trained regressor to predict the privacy settings based solely on the individual factors in the test set, and compared the results with the actual privacy settings. Neither the estimators nor the input features were changed throughout the validation. As a baseline, we used the same *random probabilistic approach* that has already been used for the mobile app domain, this time trained with the intelligent retail data from the *same training set* that was also used for the machine-learning based prediction.

The results of the analysis can be found in Table 6.6. The columns denote the condition and used input features (either the **random** probabilistic method, or the machine-learning approach using either the **IUIPC** privacy measures or our **additional** questions), the rows contain the different data types for which we predicted the permissions. Similar to the mobile phone approach, the row "all" denotes the average precision over all data types.

Similar to the mobile phone validation, the random probabilistic method can also outperform a pure random method (whose precision would be about 50%) with a precision of 57.7% in this domain. Nevertheless, it is outperformed by the machine learning-based prediction ($M_{IUIPC} = 69.1$, $M_{Additional} = 67.6$). The prediction based on the IUIPC questionnaire (12 questions) performs best, although good results can also be achieved using our additional questionnaire (two questions) if the amount of questions to be answered by the user and thus the user burden has to be decreased further. Best results can be achieved for the *Amount* permission ($M_{IUIPC} = 87.3$, $M_{Additional} = 87.3$). The disclosement setting for the *recently viewed* permission was

Item	Probabilistic	IUIPC	Additional
All	**57.7**	**69.1**	**67.6**
Name	60.0	73.6	71.8
Birthday	55.4	60.0	56.4
Address	48.1	61.8	59.1
Income	50.9	59.1	54.5
Nutrition	53.6	63.6	53.6
Allergies	67.2	81.8	81.8
Recent visits	61.8	78.2	78.2
Wishlist	52.7	64.6	65.5
Recently viewed	51.8	52.7	53.6
Receipt	49.0	60.9	60
Category	54.5	68.2	68.2
Price	50.9	59.1	59.1
Amount	80.0	87.3	87.3
Loyalty	60.9	78.2	77.3
Location	62.7	75.5	74.5

TABLE 6.6: Percentage of correct predictions for the random probabilistic method, and the machine-learning approach using either the IUIPC privacy measures or our custom (additional) privacy questions.

hardest to predict ($M_{IUIPC} = 52.7$, $M_{Additional} = 53.6$). Overall, the machine learning approach outperformed the probabilistic method by about 11%.

As the results seemed promising, we created a user interface called *"Retailio"* that uses our approach. The next section will give details about the UI as well as a final evaluation study.

6.7 "Retailio" privacy settings UI

The precision of the machine learning prediction has shown that such a personalized recommendation system can offer the users a set of privacy settings that is significantly more precise than a naive approach based on the data types or a generic set of privacy settings. In other domains, like the mobile app domain or the social web domain, related work has already tested whether such prediction approaches can be integrated into user interfaces, and whether this is accepted by users. However, we still have not tested whether such an approach is **accepted** by users, and whether they also **perceive** the predicted settings to be significantly better than random in the domain of intelligent retail data. For this purpose, we implemented and evaluated a privacy user interface empowered by our personalized privacy setting prediction. Based on the findings of the online study, we decided to create a user interface which:

1. allows the customer to set and monitor his retail privacy settings in one central place, and
2. supports the user during his decision, by utilizing the machine learning techniques described in the former sections

We therefore implemented a mobile website that gives a clear overview on the settings, and that offers a privacy assistant to set the privacy settings automatically.

The detailed workflow of the UI is denoted in Figure 6.2. When the customer opens the website for the first time after registration, he is offered a privacy assistant (see Figure 6.3, upper left) which asks the 12 questions of the IUIPC questionnaire.

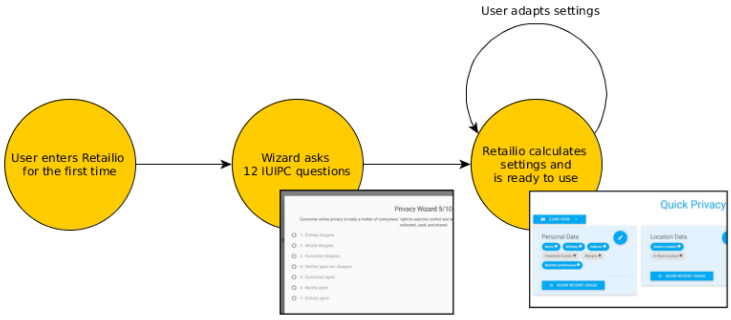


FIGURE 6.2: Typical workflow of Retailio during first use.

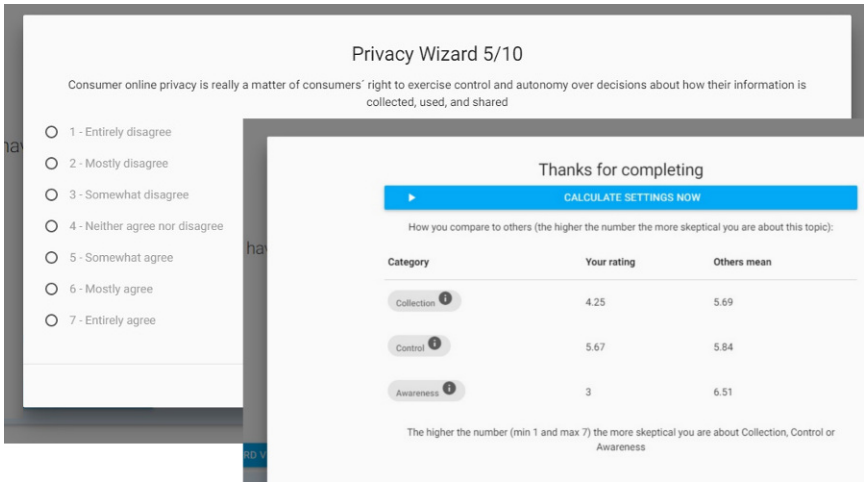


FIGURE 6.3: Retailio's privacy assistant: IUIPC questions (upper left) and results page (lower right).

After the survey is finished (typically 2-3 minutes), the customer is presented the results (mean scores) of the questionnaire along with the typical mean scores of other customers (Figure 6.3, lower right). When clicking on "calculate settings now", the privacy assistant uses the ridge regression estimators (see Section 6.5) to predict the privacy settings tailored to the customer. From that point on, Retailio is initially set up and ready to use. The customer can still manually enable or disable single permissions on the main screen (See Figure 6.4), or re-run the assistant at a later point in time.

6.7.1 Evaluation

In order to evaluate the final design of *Retailio*, we performed a lab study with 24 participants from the university context. As stated in the introduction, there is currently no system that offers the customer a user interface to set his retail privacy settings. Therefore we could only evaluate the attractiveness of the Retailio UI as it is; there is no baseline interface that we could use as a comparison. Nevertheless, we

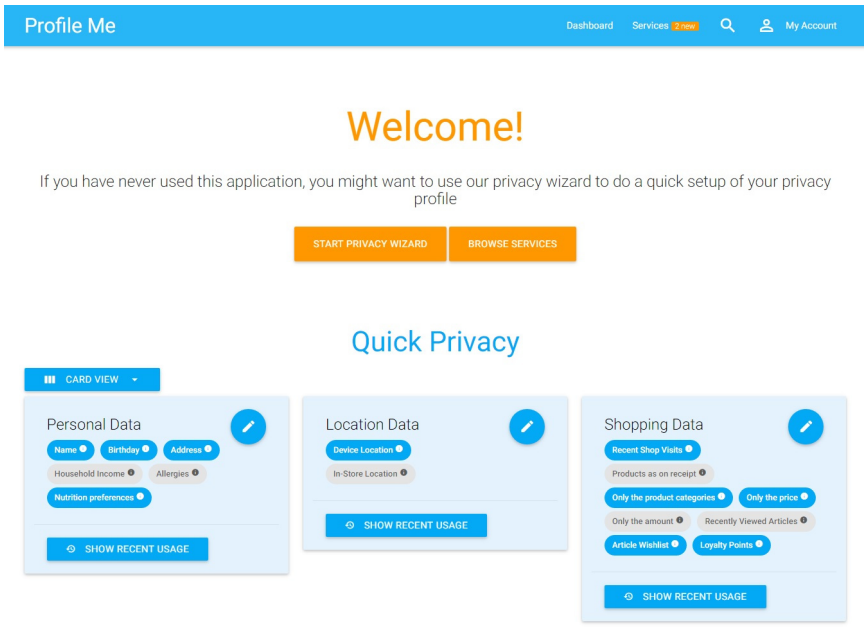


FIGURE 6.4: Main screen of Retailio.

were able to evaluate the prediction of the privacy assistant as a comparative study against the probabilistic approach as the baseline.

The goal of the evaluation was to check in a realistic scenario:

- whether the idea of using a privacy assistant is accepted by the users
- if the predicted settings of the privacy assistant are useful
- how well Retailio is rated in terms of user experience and performance
- if there are still some points for improvement

All studies were conducted remotely using a Teamviewer session. The participants accessed the Retailio website remotely over their web browser. Although the UI was the same, we tested two different conditions *for the prediction*: The first condition used the *ridge regression estimator* in the privacy assistant; the second one used the *probabilistic estimator*. Just as in Section 6.5, the second condition therefore forms the baseline condition. The order in which conditions were used was shuffled using a Latin square. The procedure was the same for both conditions: After filling out a questionnaire containing general information (gender/age etc.), the subjects were given a link and the account data for the Retailio website. After logging in, they followed the typical workflow as depicted in Figure 6.2: First, the privacy assistant was used to do an initial setup of the retail privacy settings. After that, the user reviewed the predicted settings on the main page, and changed incorrectly predicted settings. When the user finished the editing process, the procedure ended with a subjective rating of the prediction on a 10-point scale (1 = not at all accurate, 10 = very accurate). The participants were invited to a second meeting seven days after the main experiment. We chose this timespan as it was short enough so that people's privacy preferences would not change in the meantime, but long enough so that they would not remember the settings from the previous appointment. The procedure was the same as in the first meeting, this time with the other condition. The second and last

Variable	Question
General_privacy	I like the idea of privacy management in general (being able to individually set your own settings)
Privacy_assistant	I like the idea of a privacy assistant that helps me to set my permissions
Prefer_to_manual	I would prefer a privacy assistant over a manual setting
Prefer_predefined	I prefer to use predefined privacy profiles
Trust	In general, I trust the conditions and privacy statements of companies

TABLE 6.7: Additional questions asking for general attitude toward privacy assistants, privacy settings and general trust in companies' privacy policies.

meeting ended with an attrakdiff questionnaire [150] as well as additional questions about the general attitude toward privacy assistants, privacy settings and general trust in companies' privacy policies as described in Table 6.7 on a five-point ordinal scale from strongly disagree to strongly agree.

In addition to the questionnaire results, we recorded the number of changes made by the user after finishing the privacy assistant in each of the two conditions.

6.7.2 Results

We first analyzed the data on the number of changes made by the user. As a test on normal distribution failed, we used a Wilcoxon signed ranks test, which is a non-parametric test to analyze interval or ordinal data of two populations. As the results show, the settings predicted by the machine-learning-based privacy assistant ($M_{changes} = 4.35, SD = 2.76$) were changed significantly less often ($Z = 2.891, p = 0.004$) compared to the control condition using the random probabilistic approach ($M_{changes} = 5.6, SD = 2.8$).

The subjective rating for each condition (10-point scale, 1=worst, 10=best) gave us two sets of ordinal data for the two conditions. Tests on normality failed; therefore, we compared the results again using a Wilcoxon signed ranks test, as it is the preferred statistic for this kind of ordinal data. Also here, the users significantly ($Z = 2.331, p = 0.02$) preferred the machine-learning-based settings ($M = 7.05, SD = 1.5$) to those of the probabilistic privacy assistant ($M = 6.1, SD = 1.48$).

The additional questions (see Table 6.7) were proven to be normal-distributed this time using a Kolmogorov-Smirnov and Shapiro-Wilk test. Therefore we were able to use a one-sample t-test with a test value of 3 (mean of the five-point scale) for the analysis. The results can be found in Table 6.8. According to the results, with high significance, people like the idea of managing their privacy settings themselves ($t = 17.61, p < 0.005$) in general and also by using a privacy assistant for this task ($t = 10.672, p < 0.005$). Privacy assistants are preferred to manual settings ($t = 4.27, p < 0.005$). In general people do not trust the privacy statements and regulations offered by companies ($t = -3.58, p = 0.002$), highlighting the need for a custom privacy management tool like Retailio. We were not able to prove any trend in whether the subjects prefer to use pre-defined privacy templates instead of setting every single permission themselves, although there is a slight lean towards individual settings rather than privacy templates ($t = -0.195, p = 0.84$).

The results of the attrakdiff questionnaire at the end of the experiment are posted in Figure 6.5. An average user interface would have a neutral pragmatic and hedonic

Variable	mean	T	p
General_privacy	4.75	17.62	<0.005
Privacy_assistant	4.45	10.72	<0.005
Prefer_to_manual	3.70	4.27	<0.005
Prefer_predefined	2.95	0.195	0.85
Trust	2.35	-3.58	.002

TABLE 6.8: Statistical results for the additional questions.

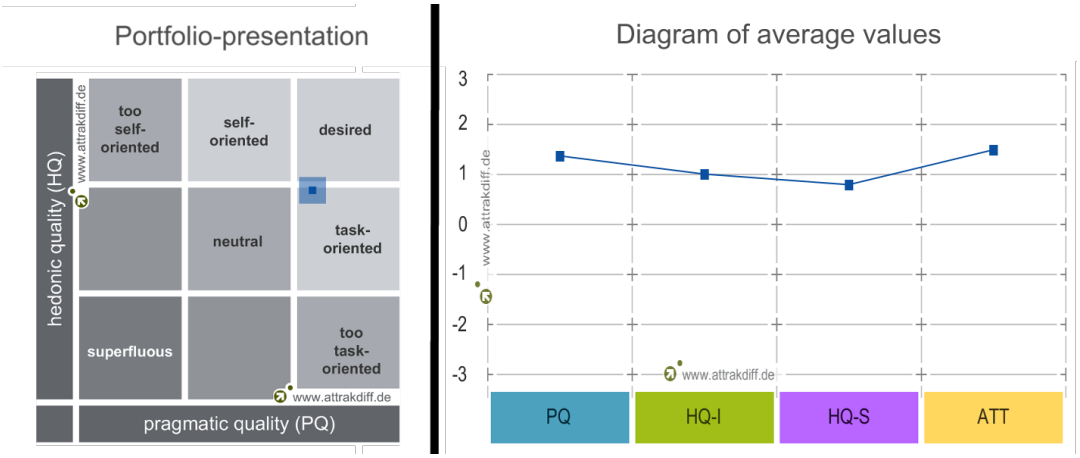


FIGURE 6.5: User experience results of Retailio: Portfolio presentation for the pragmatic and hedonic quality (left), and detailed average values of the four attrakdiff measures pragmatic quality (PQ), hedonic quality regarding stimulation (HQ-S) and identity (HQ-I) and attractiveness (ATT) (right).

score (about zero) and would therefore be located in the center of the central square (neutral) of the portfolio presentation. Scores > 1 or < -1 are perceived as *above average* or *below average* respectively [150].

Retailio received a high pragmatic score ($PQ = 1.37$) which clearly attests to a usability that is above average. Although we did not put much effort into the user experience or design aspect of the UI design, we still received a hedonic quality at the border of above average ($HQ - I = 1.04, HQ - S = 0.8$). Nonetheless, the attractiveness of the UI was found to be clearly above average ($ATT = 1.49$).

6.8 Discussion

6.8.1 Precision of the prediction vs. size of the data set

As we have seen throughout the development process of Retailio, including the correlations found (Figure 6.1) and the precision of the machine learning algorithm (Table 6.6), there is a strong correlation between the IUIPC and the retail privacy settings. Although the setting prediction led to good results (about 70% correctness), we think that it is still possible to improve the prediction. Although not fully comparable to our work, researchers from other areas, like mobile app settings prediction, achieved up to 80% correct predictions [208, 206] with a large online settings database containing several million data sets. In contrast to the mentioned work, there is no large online database of retail privacy settings that we could utilize as

training data. We would like to see whether the performance improves with a larger training set.

6.8.2 User acceptance

According to the questionnaire results in the main study (Table 6.8), customers desire to have control over the data that is collected, shared and used by intelligent retail stores. The users stated in our study that they generally distrust the companies and their privacy regulations, emphasizing the urgent need for a privacy management system like Retailio. Privacy assistants are perceived as a reasonable approach to support them while making their settings. Shoppers dislike selecting all the settings manually, and prefer a privacy assistant to do all the work for them. The predictions performed by Retailio were significantly more precise than those from a simple probabilistic method. Still, we cannot state for sure whether fine-grained individual settings are the best solution for all customers. As the question *Prefer_predefined* shows, opinions are diverse: some of the subjects stated they preferred pre-defined privacy setting templates, while some liked to be able to adapt every single setting, as offered by Retailio. A different approach to Retailio could use a finite set of privacy profile templates, and use (maybe shorter) questionnaires to select one of the templates, as is done in related work on Facebook privacy settings [272]. To sum up, we can say that a concept like Retailio is accepted.

6.8.3 User interface design

We took an iterative approach when designing Retailio, starting from background research, to a user study, checking for correlations that could be utilized as a basis for machine learning, and ending up with a proof-of-concept UI that implements our approach. Although a lot of effort was put into the implementation of the UI to make it as convenient as possible, we did not conduct an in-depth design process, including design thinking and the design and evaluation of several layouts. As the attrakdiff results show, the UI is indeed perceived as convenient on one hand; on the other hand, there is some space for improvement in the hedonic quality, i.e. the user experience as such when using the interface. Especially the stimulation measure ($HQ - S$) could be improved, meaning the interface could be designed to be more eye-catching and interesting. We would like to go through this process of designing an advanced UI, involving all the steps that are needed for a design process, in future work.

In a second step, we want to bring Retailio to customers, connecting it with an intelligent retail store like Amazon Go. We would like to explore in an in-the-wild study whether Retailio will be used in practice, how well the prediction performs with a large user base, and how useful such an approach is perceived to be by the customers.

6.9 Conclusion

New intelligent retail stores like Amazon Go make it clear that we are on the verge of brick-and-mortar stores becoming more comfortable, intelligent, customer-sensitive and individual. On the other hand, the increasing comfort and individualization comes with a need for a higher amount of individual customer data. Although some accept giving away their data for advanced customer services, not all customers want to share all their data with retail companies; sometimes they want to share

only part of it. We implemented a system called Retailio, which gives shoppers control over, and an overview of their personal shopping data, and offers a privacy assistant to automatically set up an individual initial privacy profile. We did some background research on which data is recorded in intelligent retail stores, and did an online user study to capture how far the personality and privacy awareness of a customer correlates with the desired data disclosure settings (retail privacy settings). Machine learning has been used to build a privacy assistant for Retailio, which creates the initial privacy settings profile after the user answers some simple questions. The study results show that customers have a strong mistrust of retail companies' privacy settings and a need for control over their personal data. The privacy assistant concept used in Retailio was accepted and the results of the prediction were perceived as useful. Nevertheless, our research brought some different promising approaches as well as chances for further improvements to light, which could make Retailio even stronger in a future version.

Chapter 7

Cross-domain privacy setting prediction

As we have seen in the last chapters, one way to recommend privacy settings is to use context factors or earlier privacy decisions in the same domain (for example the same social network website) as an input to recommend privacy settings. Sometimes, there is no information available about previous privacy decisions which can be used for the prediction, also known as the cold start problem [222]. In Chapters 4 to 6, we discussed alternative ways of using individual measures as an input for the prediction, solving the cold start problem, if no previous privacy decisions are available. The privacy measures can be captured using a questionnaire or inferred from blog/social network entries of the user. However, sometimes these two data sources are also not available. Still, in some cases, the user has already used other systems where privacy settings had to be selected. These could be suitable as an input for the prediction, also known as *cross-domain user modeling*. Although single-domain recommender systems should be preferred due to their higher prediction precision, if available [282], cross-domain recommender systems have the advantage that they are able to predict settings for more than one domain, leading to increased engagement and satisfaction of the user [5].

Cross-domain recommenders have been very succesful in transferring product recommendations between multiple domains, for example for recommending books based on the preferred movies of the user [345], or for recommending music based on places of interest [111]. Another approach by Dominikus Heckmann called *Ubiquitous User Modeling* [152] uses ontologies to build a generalizable user model that is applicable for multiple domains and scenarios, allowing use of the user model in different situations. For example, when the user books a flight at home, her personal preferences and personal data (such as flight time, airport etc.) are stored within the ubiquitous user model, so that it can later be used at the airport to customize the navigation to the gate according to the time left until the flight, taking either a relaxed and interesting route through some stores, or taking the most direct route if the user is about to miss the flight. Although such a ubiquitous user model is very powerful and also offers rudimentary options for defining the privacy settings of the collected data, such as an informal privacy level, ownership, purpose, and a retention date, it cannot propose privacy settings that can be directly applied for example on a social network website, as they are too abstract and not domain-specific for that purpose. Cross-domain recommender systems that do not rely on an ontology and that are tailored especially towards privacy settings have so far, to the best of our knowledge, not been a subject of research. In this chapter, we took four domains as an example, to find out whether and how well the privacy settings of one domain can be predicted using the privacy settings from one or several other domains.

As we have shown in earlier chapters, privacy decisions are not ultimately binary. In social network posts, a user might not just hide or show the complete post, but he might want to take a middle road and hide only the post image or comments, while still sharing the post text with a certain friend group. Furthermore, different domains have different privacy options to set, which makes it hard to directly compare the privacy settings per se. We therefore use *privacy levels* in our prediction. A privacy level describes how important data privacy is for the user on a continuous scale, allowing a comparison of the privacy levels between different domains. The privacy levels can be resolved to concrete privacy settings after a prediction, depending on the domain (see Section 2). In our work, we discuss two different granularities of privacy levels: first, *mean domain privacy levels* describe an average privacy level over all privacy levels for a user in a specific domain, independent of context factors. There is *exactly one* mean domain privacy level per domain. In contrast to that, there are multiple *context-based privacy levels* for a domain, one for each combination of each context factor instance (for example, there is one context-based privacy level for the post topic "family affairs" (context factor 1) in combination with the recipient group "school friends" (context factor 2). Currently, social media or location sharing services offer users a "one size fits all" solution for their users, where everything is set to a specific default value at the beginning. Using the user's mean domain privacy level, the service could already tailor all privacy settings to be more restrictive if the user has a high mean domain privacy level for that domain, or use a looser set of default privacy settings if the mean domain privacy level is low. Using the *context-based privacy levels*, one could tailor the privacy settings even better to the user, by providing different privacy settings for different contexts, for example when a new post about "family affairs" has to be shared with "school friends".

In this chapter, we will build upon our previous work on deriving privacy settings using context factors and individual factors (such as user personality or privacy attitude) described in Chapters 4 to 6. Based on the user studies presented in this chapter, we want to compare the four different domains of privacy levels for *social networks*, *location sharing*, *intelligent retail data*, and *mobile apps permissions* regarding how privacy levels for each of those domains can be predicted using the privacy levels from the other domains, and find to what extent the usage of context-based privacy levels, using different values for each context factor, plays a role in this context. In contrast to previous work, we will **not** use context or individual factors as a source for the prediction. Instead, we will predict the privacy settings of a domain using the privacy settings from another of the aforementioned domains. In our work, we will investigate which of the two mentioned granularities, mean domain privacy levels or context-based privacy levels, work best for predicting the mean domain privacy levels and the context-based privacy levels, which domains and, inside a domain, which of the privacy levels should be used for the prediction and how precise a prediction can be.

7.1 Cross-domain user modeling for privacy settings

As earlier chapters have shown, privacy decisions are influenced by several context factors and individual factors based on the user's desires and behavior. Whereas the individual factors mostly depend on the personality and privacy attitudes of the user that can be captured by the respective questionnaires like the big five personal [78] inventory or the IUIPC information privacy scale [221], the context factors differ

between the domains. For location data and social network data, those context factors are the group of recipients for the SN post or location, and the topic of the post (e.g. "family affairs") or occasion when sharing the location (for example "sports events"). For intelligent shopping data, context factors are the receiving group (e.g. the retailer, family and friends, or third parties like a marketing organization). Lastly, for mobile apps, the category of the app (like "navigation app" or "messenger") as well as the type of permission (e.g. "access contacts" or "access microphone") play the most important role for the privacy decision apart from individual factors.

However, although context and individual factors, and their use for predicting privacy levels, have already been well researched, inferring privacy levels between domains raises several new challenges for a prediction. We still don't know whether the context factors of one domain also have an impact on the privacy levels of another domain; for example, whether the choice of privacy levels for the different occasions in a location sharing scenario allow one to infer the privacy levels for the different stakeholders in an intelligent shopping data scenario, or whether it is sufficient to take the average location sharing privacy level for that user to do so. Furthermore, the actual privacy options differ between domains: For example, for a social network post, the user can decide to hide parts of the post like image content or comments, or hide it from the news wall, to increase his privacy, whereas location privacy can be tuned by reducing the precision of the shared location (for example by sharing only the city instead of the exact location). To conclude, at present, we do not know which context factors of one domain are important predictors for another domain; furthermore, the value set of privacy levels differs (in terms of answer options as well as number of possible answers) between domains, making it hard to create a direct mapping of privacy levels between them.

We performed two iterative studies using a bottom-up approach: In the first study (**"exploratory study"**), we performed regression analyses on the generic privacy levels for each domain and analyzed the regression coefficients,

1. to find out the domains containing privacy settings that are suitable regression coefficients ("input variables") for the regression of each of the four target domains, and

2. to identify which context-based privacy levels are potential candidates for improving the precision of the prediction of

   (a) the generic privacy level of the target domain and

   (b) the context-based privacy levels of the target domain

In the second study (**"validation study"**), we validate the choice of the context-based privacy levels using a fresh data set and compare the regression precision using either

1. the generic privacy levels of the suitable domains as identified in 1 or

2. the context-based privacy levels that have been identified as suitable in 2,

for predicting the generic privacy levels for each domain, as well as the context-based privacy levels.

In more detail, we pursue four different kinds of predictions, as depicted in Figure 7.1. For the *mean-based regression analysis (MGR)*, we work only on the top level, including the *mean domain privacy levels*, neglecting any context information, i.e. we are trying to predict mean domain privacy levels using the mean domain privacy

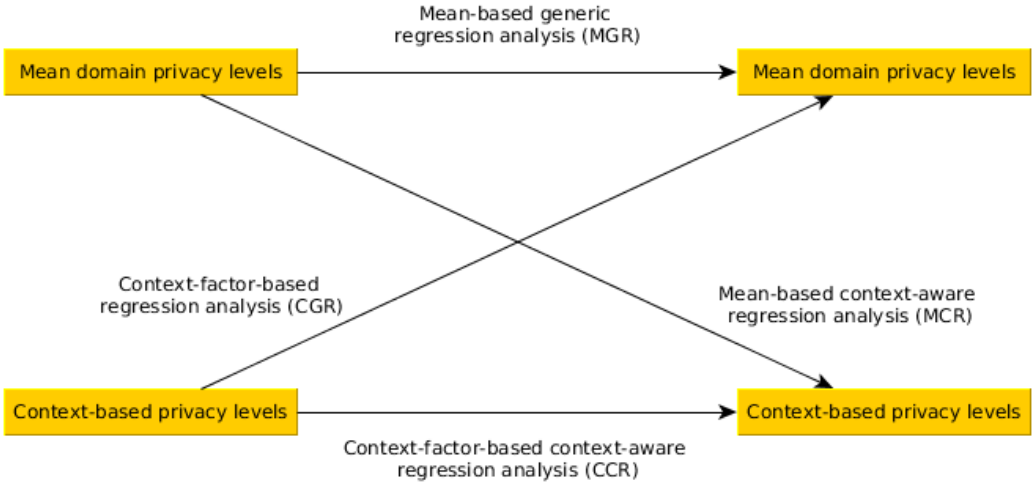


FIGURE 7.1: Planned regression analyses.

levels of the other domains. For the context-factor-based regression analysis (CGR), we use context-based privacy levels on the input side, to predict the mean domain privacy levels on the output side. Conversely, the mean domain privacy levels could also be used as coefficeints for a regression of the context-factor-based privacy levels (*Mean-based context-aware regression analysis (MCR)*). Finally, the context-factor-based context-aware regression analysis (CCR) uses context-factor-based input to predict context-factor-based privacy levels. Our ultimate goal is to find out which of the approaches works best for predicting the mean domain privacy and the context-factor-based privacy levels, which domains and, inside the domains, which context-factor based privacy levels should be used; and what standard error can be achieved.

7.2 Exploratory study

The exploratory study and the validation study were conducted as an online study using a local LimeSurvey[1] installation at our institution. The participants were recurited using an online recruiting platform called Prolific Academic[2]. According to study results, the audience recruited through such a platform can be compared to an audience recruited through conventional methods like notice boards at a university or social network posts[53]. The participants were paid £2.10 for a succesful participation, which took 20 to 25 minutes, so that the minimum wage of £5 per hour that is required for studies on Prolific Academic is guaranteed. At the beginning of the questionnaire, the participants had to confirm that they would carefully read and answer all the questions, as not following the instructions or giving contradictory answers might lead to their participation being rejected. The actual questionnaire then asked demographic questions and whether the subject actively uses social networks and smartphones. To ensure domain knowledge of the participants, we required them to be active users of at least one social network and to own and use a smartphone. When the participant entered responses not fulfilling these criteria, we ended the survey at that point. All other participants were then asked about their

[1] https://www.limesurvey.org (last accessed: 2020-03-09)
[2] https://prolific.ac/ (last accessed: 2020-03-09)

privacy preferences in the four aforementioned domains. The order of the domains was shuffled for each participant to avoid bias in the results.

At the beginning of each block of questions, we gave the participants an introduction to the situation that we were targeting, for example "imagine you are creating a new social network post about your **hobbies**" for the social network domain. We then asked, for each group of recipients (like "friends", "family", etc.), which privacy option they would choose, according to the privacy options as presented in Chapters 4 to 6. This procedure was conducted for all combinations of

- post topic and group of recipients (social network domain)
- occasion and group of recipients (location sharing domain)
- stakeholder and data type (intelligent shopping domain)
- application category and permission type (mobile phone domain)

As privacy options, we gave the users the same options that we gave the participants in earlier studies in the respective domains (see Chapters 4 to 6 for details), namely:

- social network domain: five different privacy levels (show on timeline, show on page, hide images and comments, hide post, hide even if reshared)
- location sharing domain: the seven different location abstraction levels offered by the Google Maps API (exact location, street only, city only, province only, country only, continent only, no location)
- intelligent shopping domain: allow/deny for each *binarized granularity* (see Chapter 6) option for all combinations of data type and stakeholder
- mobile phone domain: allow/deny for each combination of app category and permission type

To assure the quality of the answers, we added four control questions in the section "location sharing" that the users had to answer exactly as stated in the task description. At the end of the questionnaire, the participants were offered a text box to enter any comments or ideas for improvements about the questionnaire. This procedure was reviewed and approved by the ethical review board of our institution.

7.2.1 Results

In total, we had 109 paticipants that completed the questionnaire; they needed on average about 23 minutes for the task. Eight result sets had to be discarded as a control question was answered incorrectly, leading to 101 viable results. The age of the participants ranged from 18 to 64 years (mean 32.39, stdev. 9.55). 58 participants were female, 43 male. 26 had already heard of intelligent retail stores, and 75 had not.

After importing the data for the analysis, we first computed several average values that will be used in the analysis later:

- For each context factor (see Chapters 4 to 6), we computed the mean privacy level for each instance of the context factor. For example, to compute the mean privacy level for the "events" occasion in the location sharing domain, we averaged the privacy levels for the "events" occasion for the different recipient groups (e.g. we calculated the average over one column or one row in the context-based privacy levels table described in Chapters 4 to 6). The averaged values will later be called **(mean) context factor privacy levels**.
- For each domain, we computed the average over all privacy levels, regardless of the context factors. These values are denoted as **mean domain (privacy) levels**.

We used the following procedure for the analysis:

1. **Context factor difference analysis**: We performed a variance analysis on the *mean context factor privacy levels* for each domain and context factor, to find out **which context factors lead to a significant difference in privacy levels**.
2. **Mean-based generic regression analysis**: We performed a regression analysis on the *mean domain privacy levels* for each domain, using the *mean domain privacy levels* of the other domains as regression coefficients, to determine **which other domains are of influence for a prediction of the privacy levels**.
3. **Mean-based context-aware regression analysis**: We performed the same kind of analysis on the *mean domain privacy levels* for each domain, using the *context-based privacy levels* of the other domains as regression coefficients, to determine **how precise a prediction on the context-based privacy levels** can be when using only *mean domain privacy levels*.
4. **Context-factor-based generic regression analysis**: Conversely, we performed a regression analysis on the *mean domain privacy levels* using the *mean context factor privacy levels* as regression coefficients, to determine which context factor instances could be of influence for the prediction. At this point, we were only interested in filtering out potential candidates, and building up hypotheses on which context factor instances could be of influence. As we were reusing the same data for multiple analyses, the reported significance values cannot be used to determine which context factor instances are significant procedures without applying alpha correction. We therefore validate the results of the exploratory study later in the validation study using a fresh data set.
5. **Context-based context-aware regression analysis**: Finally, we performed a regression analysis on the *context-based privacy levels* for each domain, using the *context-based privacy levels* of the other domains as regression coefficients, to determine **how precise a prediction on the context-based privacy levels** can be when using *context-based privacy levels*.

The results of the above analyses will be described in the next subchapters.

TABLE 7.1: Average privacy levels and tests for variance on the context factors using a Friedman test.

Domain	avg. privacy level	context factor	χ^2	asymp. sig.
Social	2.05 (41%)	topic	263.24	< 0.001
		friend group	254.64	< 0.001
Location	3.91 (56%)	occasion	154.22	< 0.001
		requestor	328.80	< 0.001
Mobile	0.57 (57%)	category	37.50	< 0.001
		permission	4.70	0.45
Shopping	0.46 (46%)	data type	44.28	< 0.001
		stakeholder	25.825	< 0.001

Context factor difference analysis

Prior to the analysis, we tested each data set for normal distribution and sphericity, in order to decide on the correct statistical test for the variance analysis (e.g. ANOVA or its non-parametric equivalent, the Friedman test). For each domain and context factor, we had at least one context factor instance for which the mean context factor privacy levels were *not* normally distributed. Therefore, we performed Friedman tests for all variance analyses. In addition to the variance analysis, we computed the mean privacy levels for each domain. As the scales have a different size, we added a normalized percentual average privacy level for each entry. The results for the analysis are shown in Table 7.1.

For the social network domain, both context factors (post topic and receiving friend group) lead to significantly different settings ($\chi^2_{topic} = 263.24$, $p < 0.001$; $\chi^2_{friendgroup} = 254.64$, $p < 0.001$), assuring that both context factors have a significant impact on the choice of the privacy levels and supporting the results of earlier chapters. The same holds for the location sharing domain, whereas the difference in privacy levels for the *requestors* is higher ($\chi^2_{topic} = 328.80$, $p < 0.001$) than for the *occasion*($\chi^2_{occasion} = 154.22$, $p < 0.001$) when the location is shared. In Chapter 4, we also found those context factors to be significant, although we found the occasion to be more important than the requestor, whereas others state the requestor to be more important [246]. In the intelligent shopping domain, the *data type* has a higher influence on the privacy level ($\chi^2_{topic} = 44.28$, $p < 0.001$) than the stakeholder requesting the data ($\chi^2_{topic} = 25.82$, $p < 0.001$). However, for the mobile app permissions, only the app category seems to lead to a significant difference in the permission settings ($\chi^2_{topic} = 37.50$, $p < 0.01$). The privacy levels for the different permission types do not differ significantly ($\chi^2_{topic} = 4.70$, $p = 0.45$). The average privacy levels for location sharing (M=2.05) and mobile apps (M=0.57) are very similar when normalized to a percentual scale (56% and 57%, respectively) and higher than the mean privacy levels for social media (M=2.05, normalized 41%) and shopping (M=0.46, normalized 46%).

TABLE 7.2: regression analyses for the mean domain privacy levels.

Target domain	coefficients	R^2	$adj.R^2$	stderr	F	sig.
Location	**all**	0.21	0.19	1.43	12.73	< 0.001
	social	0.19	0.18	1.43	22.63	< 0.001
	mobile	0	-0.1	1.59	0.02	0.88
	shopping	0.06	0.05	1.55	6.02	0.016
Social	**all**	0.22	0.19	0.61	9.02	< 0.001
	location	0.19	0.18	0.61	22.63	< 0.001
	mobile	0.03	0.02	0.67	3.00	0.086
	shopping	0.06	0.05	0.66	5.79	0.016
Mobile	**all**	0.19	0.17	0.33	11.49	< 0.001
	social	0.03	0.02	0.36	3.00	0.086
	location	0	-0.1	0.37	0.02	0.877
	shopping	0.19	0.18	0.31	22.43	< 0.001
Shopping	**all**	0.24	0.22	0.30	10.42	< 0.001
	social	0.06	0.05	0.33	5.79	0.018
	location	0.06	0.05	0.33	6.02	0.016
	mobile	0.19	0.18	0.31	22.43	< 0.001

Mean-based generic regression analysis (MGR)

To find out which domains are reasonable regression coefficients for a certain domain X, we first performed a separate regression analysis for each domain different from X, followed by a regression analysis including all domains that have a tendency to become significant coefficients (meaning $p < 0.10$). For each regression, we report the goodness of fit (R^2) and adjusted goodness of fit ($adj.R^2$) describing how well the regression curve fits the data, as well as the results (F and significance) of the ANOVA analysis, describing whether a prediction using a regression produces viable results with the given coefficients. The results are shown in Table 7.2. Note that it is typical for a regression that measures like R^2 and results of the variance analysis are the same for a regression on X using Y as a coefficent as they are for Y using X as a regression coefficient. However, to maintain readabilty and an easy comparison of the coefficients, we included both combinations in the table. All significant coefficients are printed in **bold** face.

The best regression coefficients for the location domain are the *mean domain privacy levels* from the social and shopping domains, which result in a highly significant (F=22.63, $p < 0.001$) and a significant prediction (F=6.021, $p = 0.016$), respectively. However, the mean privacy level of the mobile domain has a negative adjusted goodness of fit (R^2); the ANOVA further implies that the prediction does not generate viable results, leading to the assumption that this domain is not of use for the prediction of a mean location sharing privacy level. A similar picture can be seen for the social network domain, where the best coefficient is the mean domain privacy level of the location domain (F=22.63, $p < 0.001$). However, the shopping domain as a coefficient still produces viable results (F=5.79, $p = 0.018$), and the regression using the mobile permission privacy level has a tendency to become a viable regression coefficient (F=3.00, $p = 0.086$).

The generic mobile app permission privacy level can be predicted best using

the mean domain privacy level from the shopping domain (F=22.43, $p < 0.001$). The mean domain privacy level from the social network domain has a tendency to become a viable coefficient (F=3.00, $p = 0.086$), whereas the mean domain privacy level from the location sharing domain is of no use for this kind of prediction (F=0.02, $p = 0.877$). Lastly, the mean domain privacy level from the shopping domain is predicted best using the mean privacy level of the mobile app domain (F=22.43, $p < 0.001$), followed by the mean domain privacy level from the social (F=5.788, $p = 0.018$) and location (F=6.02, $p = 0.016$) domains which both provide a viable prediction.

Context-factor-based generic regression analysis (CGR)

For the context-factor-based generic regression analysis, we used the *mean context factor privacy levels* as regression coefficients to find out whether an increased detail level (e.g. one privacy level for each combination of context factors instead of one generic domain privacy level) can lead to an increased prediction precision in the regression. For this purpose, we first had to find out which instances of each context factor are suitable coefficients. However, as stated earlier, the significance values here cannot be seen as final (without using alpha correction or validation in a follow-up study), as the same data set is used multiple times. Later, we will validate the choice of context factor instances in the validation study using a fresh data set. Similar to the analyses described above, we select all context factor instances that have the *tendency* to become significant ($p < 0.1$), so that we do not omit any instance that might be significant within another data set, while eliminating other instances that will most likely not become significant and that would disturb the regression algorithm. Note that we excluded the "permission type" context factor, as it was found to be insignificant in the context factor difference analysis. The results can be found in Table 7.3.

Similar to the results of the generic regression analysis, the domains with the lowest standard error and the highest precision in the generic regression analysis have the highest number of significant context factor instances here. The location domain can be predicted best by the context factor instances from the social media domain, namely the privacy level for posts about events (t=2.49, p=0.015) and movies (t=1.753, p=0.083) as well as the privacy level for the friend group "school friends" (t=2.49, p=0.015). However, the privacy level for the amount of products bought in the shopping domain can also be used as a regression coefficient (t=1.672, p=0.098).

A similar picture can be seen for the location sharing domain, which can be predicted best using the context factor privacy levels from the location sharing domain, where the privacy levels of the occasions about having "food" (t=1.79, p=0.077), "traveling" (t=2.071, p=0.041) and "tech events" (t=2.203, p=0.03) are found to be suitable, together with the privacy level of the requestor groups "immediate family" (t=1.804, p=0.075) and "extended family" (t=1.803, p=0.075). From the other coefficients, only the "games" category of the mobile app domain had a tendency to be a useful regression coefficient (t=1.696, p=0.093).

For the mobile app domain, we found coefficients from different domains to be useful. Most are found in the intelligent shopping domain, namely the privacy levels of the "income" of the customer (t=1.88, p=0.064) and his "birthdate" (t=1.78, p=0.077), as well as, with high significance, the stakeholder "third parties" (t=3.818, $p < 0.001$). Using the regression coefficients from the location sharing domain, only the context factor privacy levels of two requestors, namely "close friends" (t=1.668,

TABLE 7.3: Tendentially significant regression coefficients (context factor instances) for the prediction of the domain privacy levels.

Target domain	coefficients	instance	t	sig.
Location	social - topic	events	2.49	0.015
		movies	1.753	0.083
	social - recipients	school friends	1.661	0.099
	mobile - category	-	-	-
	shopping - data type	amount	1.672	0.098
	shopping - stakeholder	-	-	-
Social	mobile - category	games	1.696	0.093
	shopping - data type	-	-	-
	shopping - stakeholder	-	-	-
	location - occasion	food	1.79	0.077
		travel	2.071	0.041
		tech events	2.203	0.03
	location - requestor	immediate family	1.804	0.075
		extended family	1.803	0.075
Mobile	social - topic	sports	2.215	0.029
	social - recipients	close friends	2.008	0.048
	location - occasion	-	-	-
	location - requestor	extended family	1.855	0.067
		close friends	1.668	0.099
	shopping - data type	birthdate	1.78	0.077
		income	1.88	0.064
	shopping - stakeholder	third parties	3.818	< 0.001
Shopping	social - topic	-	-	-
	social - recipients	immediate family	1.959	0.053
	location - occasion	-	-	-
	location - requestor	immediate family	2.067	0.042
	mobile - category	social media	1.964	0.052

p=0.099) and "extended family" (t=1.855, p=0.067) seem to be suitable. From the social media domain, the "sports" topic (t=2.215, p=0.029) and the recipient group "close friends" (t=2.008, p=0.048) are both statistically significant regression coefficients.

Lastly, the shopping domain can only be predicted by a few coefficients from different domains. From the social media domain, the recipient group "immediate family" is found to be suitable (t=1.959, p=0.053). The same coefficient "immediate family" is a significant coefficient from the context factor "requestor" from the location sharing domain (t=2.067, p=0.042). From the mobile app coefficients, the privacy level for "social media" apps seem to be suitable (t=1.668, p=0.099).

Mean-based context-aware regression analysis (MCR) and context-based context-aware regression analysis (CCR)

Both analyses try to predict the *fine-grained* context-based privacy levels. The mean-based context-aware regression analysis uses *coarse-grained* mean domain privacy levels as a source for the prediction, whereas the context-based context-aware regression analysis uses *fine-grained* context-based privacy levels as an input. The procedure used here was the same that we employed for the CGR method, i.e. we performed a regression analysis on all input variables, and selected those with a p-value < 0.1 for the validation study. Taking all four domains into account, we have a total of about 100 context-based privacy levels, with up to four viable coefficients each for the MCR, and again up to about 100 coefficients for the CCR. For each of those combinations, we would have to report t and significance values. For the sake of brevity, we will not report and discuss all viable coefficients here, but report the results of the final regression analysis in the validation study in Table 7.6.

7.2.2 Discussion

Context factor difference analysis and choice of context factors

For most of the context factors that we chose according to related literature, we confirmed their significant impact on the choice of privacy settings in the *context factor difference analysis*. Especially for the social network domain, both the recipient group (or friend group) as well as the topic of the post are very important context factors. For the location sharing domain, the requestor seems to have a larger impact on the privacy setting than the actual occasion, which supports earlier work [31, 74] that came to a similar conclusion, that the requestor and occasion are the most important factors when the location is shared. Whether the requestor or occasion is more important, differs between earlier publications: Some found the requestor to be the most important factor [31, 74] whereas the work presented in Section 4.1 found the occasion to have a higher influence. According to our results in the intelligent shopping domain, the stakeholder requesting the data is also a significant context factor, but is less important than the data type (for example viewed products or in-store movements) that is requested. We assume that the high diversity of data in the intelligent shopping domain might cause this effect, as the need for privacy differs more for data types like household income or in-store movements, which might be considered more private than one's birthday or loyalty points earned throughout the shopping processes. However, as other studies on the importance of context factors also led to different results in other domains, the results have to be validated in further studies. In the location sharing domain, the data type is always the same, and therefore yields a similar perceived criticality when shared unintentionally. This

leads to the assumption that the importance of the "data type" as a context factor might rely on the diversity or number of data types (for example whether data in the domain consists only of GPS locations, or whether there is demographic data, financial data and location data within the domain), or both, which should be further investigated in future research. Interestingly, although the current privacy user interfaces in smartphone operating systems (Android or iOS) are tailored towards setting the permission individually for each permission type, our results indicate that the difference in privacy levels between permissions is not significant. In contrast to that, the category to which the app belongs has a strong influence on the privacy level ($p < 0.001$). It seems that either the users trust apps from a certain category and grant the permissions, or they do *not* trust that *kind* of app and deny all of them. If the results can be supported by future studies, smartphone suppliers might want to redesign their permission UI, and include the app category as an option to let the user decide whether an app from that category should receive all permissions, or whether only some of them should be granted. The average privacy levels for location sharing and mobile apps are higher than for the other two domains, signaling that both location sharing and mobile app permission settings are perceived as more critical, or the recipient groups less trustworthy than for the two other domains. Apart from the permission type in the mobile app domain, all context factors have been proven to have a highly significant impact ($p < 0.001$) on the permission settings, supporting earlier work that relies on context factors for recommending privacy settings [246].

Generic regression analysis

According to the results, we have two clusters of domains that can profit from each other for a prediction using a regression: the location sharing domain and social media domain privacy levels seem to be good regression coefficients for each other and form the *"location-social cluster"* on the one hand, whereas mobile app settings and intelligent shopping privacy levels form another cluster, later called the *"shopping-mobile cluster"*, that allow a good prediction of each other's privacy levels. For location sharing and social media, if the data of the other domain is not available, the privacy levels from the shopping domain, and to some extent also from the mobile app domain (for social media), can be used. But if the data of the other domain within the cluster is available, adding the coefficients from the other domains does not reduce the standard error for the location-social cluster. So if the mean domain privacy level of the other domain inside the cluster is available, the privacy levels from the other domains can be omitted, according to our results.

For the shopping-mobile cluster, the situation is not that clear. Although using the mean domain privacy level of the cluster partner as a coefficient leads to the lowest standard error compared to the other domains, the social and location privacy levels are very good alternatives, especially for the intelligent shopping domain, where both coefficients are found to be significant. Therefore, combining all three domains leads to the lowest standard error for the intelligent shopping domain, although the two additional domains outside the cluster reduce the standard error only from 0.31 to 0.30. For the mobile app domain, adding coefficients other than the mean privacy level of the cluster partner cannot improve the prediction precision; therefore, if the privacy level from the intelligent shopping domain is available, all other data should be omitted for best results. If this is not the case, the privacy level from the social media domain also allows a prediction slightly better than random.

On the other hand, data from the location sharing domain is useless and should not be used for this domain.

Interestingly, the two clusters always contain the two domains that have a similar granularity, meaning they have a similar number of privacy levels (see Chapters 4 to 6). Whereas it seems clear that it is hard to use a binary scale from the shopping-mobile cluster to predict a more fine-grained scale from either the social media or the location sharing domain, this should not be the case for the other way around. However, the context-based location and social privacy levels were also of no use to enhance the regression for the mobile app domain. Also for the intelligent shopping domain, the decrease in standard error is small. We therefore suppose that the existence of those two clusters is not a main product of the difference in their scales, but is caused by some other factor, like the type of occasion when the decision is made (for example whether it is made incidentally on the go for mobile apps or during shopping inside an intelligent retail store, vs. as a main task during a leisure activity for the two others) or the type of privacy (privacy from companies like app manufacturers or retailers on one hand, and friends or family members on the other). Which factors finally led to the clustering of domains, and which other clusters exist, should be further investigated in future work.

Context-factor-based generic regression analysis

In general, we can see that, similar to the generic regression analysis, the domains within the corresponding cluster in general produce the most, and most significant, context-factor-based coefficients (CFB-coefficients), supporting the correctness of the analysis. Taking a look at the most significant CFB-coefficents, we can see that the coefficient "third parties" in the "mobile apps" target domain has the highest significance of all of them. However, other stakeholders, like the retailer, do not even have tendency to become significant ($p > 0.1$). This fact leads to the assumption that the trust that consumers put in app manufacturers is comparable to the trust they put in third parties like marketing companies, and not like the trust they put in a retailer. This is interesting, as app manufacturers and retailers are both the direct providers of the service the costumer requests, unlike marketing companies, which usually do not offer a direct advantage for the customer. A possible explanation for this circumstance might be that the mere size, brand awareness or privacy image of a company is a key indicator for trust in terms of privacy, rather than the service quality or the benefit from the service. This assumption is supported by other significant CFB-coefficients of the shopping domain, like one's birth date and household income. Both data types are on average perceived as very sensitive by most customers, and therefore are shared rarely, indicating that the trust in mobile app developers and the will to offer them access to app permissions is relatively low. Interestingly, the other CFB-coefficients (outside the location-social cluster) that have been found useful for the mobile app domain indicate a less privacy-sensitive behavior. Posts about sports, and for close friends or the extended family, are usually not very restricted. However, taking a look at the results of the generic regression analysis, including these other coefficients in the prediction actually reduces the precision and increases the standard error; we therefore assume that they are just statistical artifacts.

For predicting the social media privacy level, both the location sharing settings from the immediate and extended family are found to be useful, meaning that the privacy levels used when sharing the location with members of the family are similar to those used in social networks. Considering the mean domain privacy levels, we can see that location sharing privacy settings are typically stricter than settings for

social media posts. Furthermore, users usually have relatively loose privacy settings for their family members (see Chapter 4). However, as the location sharing domain is stricter in general, we assume that the loose settings in this stricter domain can be compared to an average privacy level in the social media domain, making it a good regression coefficent. Similar to this, more private occasions like traveling or preparing meals together ("food") have been found to be good CFB-coefficients.

For the location sharing domain, most viable coefficients have their origin in the social media domain. The best coefficient is the privacy level of the posts with topic "events", most likely because events are the occasion where users typically share a location. Also "movies", e.g. social network posts about watching movies together or going to the cinema, are common occasions when a location is shared, making it the second most important CFB coefficient. We also found the recipient group (friend group) "school friends" to be viable, although the p-value is relatively high. In our opinion, locations are usually shared when a user does something interesting in her life, like attending events or having a meal at an expensive restaurant. The things in your life that you want to tell your school friends in order to improve your image are typically the same things. The same might hold for the amount of items that you bought at a shop. Therefore we think both privacy levels correlate to the location sharing domain privacy level, and hence are good CFB-coefficients for the regression.

Lastly, in the shopping domain, we have one CFB-coefficient from each of the other domains. The results indicate that the recipient or friend group "immediate family" both from the social media and location sharing domains, as well as permission settings for social media apps, are good coefficients. As stated before, most users use relatively loose privacy settings for their immediate family. Furthermore, social media apps require a lot of different permissions in order to be fully functional, like access to stored images or the location, which may lead users to grant them these permissions. So overall, the shopping domain seems to be a domain where customers feel confident when sharing their data, because they either do not see much harm in oversharing, or because their trust in retailers is relatively high. The lower average domain privacy level supports this assumption. In the next step, we will validate the choice of the aforementioned regression coefficients (Tables 7.3 and 7.2 and the candidates from the MCR and CCR method) using a fresh data set, which will be done in the validation study in the next section.

7.3 Validation study

In the exploratory study, we had the goal to get a first impression of how accurate the prediction of the mean domain privacy levels can be when the mean privacy levels of the other domains are used, which domains are useful for a prediction, and which context factors (like the recipient of the data or the occasion, see Chapters 4 to 6) and mean domain privacy levels (i.e. the average over all privacy levels of a domain, see section 7.1) are potential candidates for the prediction of the domain and context-based privacy levels. In the validation study, we will validate the results from the exploratory study, especially how well the regression performs with the selected coefficients, and which of the outlined approaches (MGR or CGR for predicting mean domain privacy levels, and MCR or CCR for the context-based privacy levels) performs best.

As we found out in the context factor difference analysis, the permission type does not have a significant influence on the permission level. We have therefore

excluded this context factor in the validation study. Apart from this change, the procedure for collecting data was similar to that of the exploratory study.

We had 117 participants in the validation study, out of which 11 were discarded, as they answered a control question incorrectly, resulting in 106 valid records. The participants were aged between 18 and 71 years (average 32.67) and needed on average about 27 minutes for the task. 52 participants were female, 54 male. All of them use a smartphone.

7.3.1 Results

In the validation, we again performed the regression analyses from the validation study with the newly collected data set. Instead of using all coefficents for the regression, we used exactly the regression coefficients that were found viable in the exploratory study. To be more precise, we again performed:

- **Mean-based generic regression analysis (MGR)** trying to predict **mean domain privacy levels** from a domain using **mean domain privacy levels** from other domains
- **Context-factor-based generic regression analysis (CGR)** trying to predict **mean domain privacy levels** from a domain using **context-based privacy levels** from other domains
- **Mean-based context-aware regression analysis (MCR)** trying to predict **context-based privacy levels** from a domain using **mean domain privacy levels** from other domains
- **Context-factor-based context-aware regression analysis (CCR)** trying to predict **context-based privacy levels** from a domain using **context-based privacy levels** from other domains

In the following section, we will first present the results from the statistical analyses. The interpretation of the results follows in the discussion section later.

Mean-based generic regression analysis (MGR)

As stated before, we use the mean domain privacy levels from the exploratory study that have been found to be viable (e.g. $p < 0.1$). The results can be found in Table 7.4. Remember that the MGR is based solely on mean domain privacy levels, meaning it uses mean domain privacy levels from other domains as an input to predict the mean domain privacy level of the target domain. Context factors or context-based privacy levels are not used.

For the location sharing domain, the social media mean privacy level allows the most precise prediction (stderr=1.27). Both the shopping (stderr=1.37) and the mobile (stderr=1.35) mean privacy level result in a higher standard error in the analysis. Compared to the experimental analysis, the precision and R^2 values of the value pair "location sharing" – "mobile apps" have significantly improved. In the social media domain, including all other mean domain privacy levels led to a standard error of 0.498 on the seven-point privacy level scale for this domain. Using only single mean domain privacy levels as an input, the location domain performs best (stderr=0.51), followed by the shopping (stderr=0.54) and mobile (stderr=0.57) privacy levels. The mean privacy level of the mobile app domain can be predicted best by using either

TABLE 7.4: Validation study results for the MGR analysis.

Target domain	coefficients	R^2	$adj.R^2$	stderr	F	sig.
Location	**all**	0.27	0.25	1.26	14.43	< 0.001
	social	0.24	0.23	1.27	36.61	< 0.001
	mobile	0.14	0.14	1.35	17.53	< 0.001
	shopping	0.12	0.12	1.37	14.73	< 0.001
Social	**all**	0.30	0.27	0.50	14.42	< 0.001
	location	0.24	0.23	0.51	23.61	< 0.001
	mobile	0.06	0.05	0.57	7.11	0.009
	shopping	0.16	0.15	0.54	19.83	< 0.001
Mobile	**all**	0.15	0.14	0.27	9.27	< 0.001
	social	0.06	0.06	0.28	7.11	0.009
	location	0.14	0.14	0.27	17.53	< 0.001
	shopping	0.14	0.13	0.27	16.91	< 0.001
Shopping	**all**	0.25	0.23	0.27	11.30	< 0.001
	social	0.16	0.15	0.28	19.83	< 0.001
	location	0.13	0.12	0.29	19.83	< 0.001
	mobile	0.16	0.15	0.28	19.60	< 0.001

the mean privacy level of the intelligent shopping (stderr=0.27) or location sharing domain (stderr=0.27). Using both viable domains from the experimental study (social and shopping) results in the same standard error of 0.27. The social media domain performs slighty worse, resulting in a stderr of 0.28. Lastly, in the intelligent shopping domain, all single domains produce similar results. Both the social media and mobile domain privacy levels allow a prediction with a standard error of 0.28; the location sharing domain is only slightly worse (stderr=0.29). Using all three domains together allows us to reduce the standard error to 0.27.

Context-factor-based generic regression analysis (CGR)

Also for the CGR, we used only the context factor instances that were found to be useful in the exploratory study, together with the mean domain privacy levels of the respective domain(s). The results can be found in Table 7.5. Note that for some domains (for example the mobile domain for predicting location sharing), none of the context factor instances was found to be suitable in the exporatory study. Those domains are marked using "-" in the table.

In the location sharing domain, using all viable context factor coefficients leads to the smallest standard error (stderr=1.20) for this domain, followed by the "topic" context factors of the social media domain (stderr=1.22). However, using the "recipient" context factors leads to a higher standard error (stderr=1.43); therefore, combining them with the "topic" coefficeints leads to a higher standard error (stderr=1.23) than using the "topic" coefficients alone. For the social media domain, equally good results can be achieved using only the location sharing coefficients (stderr=0.51) or a combination of all viable coefficients. If only data from the mobile app domain is available, a prediction with a standard error of 0.57 can be achieved. Data from an intelligent retail store did not provide any significant coefficient in the exploratory

TABLE 7.5: Validation study results for the CGR analysis.

Target domain	coefficients	R^2	$adj.R^2$	stderr	F	sig.
Location	all	0.36	0.32	1.20	7.95	< 0.001
	social - all	0.31	0.27	1.23	8.9	< 0.001
	social - topic	0.33	0.30	1.22	15.12	< 0.001
	social - recipients	0.19	0.18	1.43	22.63	< 0.001
	mobile - category	-	-	-	-	-
	shopping - data type	0.18	0.15	1.34	5.44	0.001
	shopping - stakeholder	-	-	-	-	-
Social	all	0.30	0.24	0.51	5.22	< 0.001
	mobile - category	0.06	0.06	0.57	7.11	0.009
	shopping - data type	-	-	-	-	-
	shopping - stakeholder	-	-	-	-	-
	location - all	0.26	0.22	0.51	5.78	< 0.001
	location - occasion	0.24	0.22	0.51	10.57	< 0.001
	location - requestor	0.25	0.22	0.51	8.38	< 0.001
Mobile	all	0.20	0.15	0.27	4.018	0.001
	social - all	0.11	0.08	0.28	4.09	0.009
	social - topic	0.11	0.09	0.27	6.11	0.003
	social - recipients	0.06	0.05	0.28	3.53	0.033
	location - occasion	-	-	-	-	-
	location - requestor	0.14	0.14	0.27	17.53	< 0.001
	shopping - all	0.15	0.10	0.27	3.42	0.007
	shopping - data type	0.15	0.11	0.27	4.31	0.003
	shopping - stakeholder	0.14	0.13	0.27	16.91	< 0.001
Shopping	all	0.23	0.20	0.27	6.07	< 0.001
	social - topic	-	-	-	-	-
	social - recipients	0.16	0.15	0.28	19.83	< 0.001
	location - occasion	-	-	-	-	-
	location - requestor	0.12	0.12	0.28	19.83	< 0.001
	mobile - category	0.16	0.15	0.28	19.60	< 0.001

study and therefore cannot be more precise than a prediction using the MGR approach. For the mobile app domain, using the coefficients from the location, the shopping domain or the "topic" coefficients from the social media domain results in a stderr of 0.27, equal to that of the MGR method. Using the "recipient" coefficients, or using them together with the other social media coefficients, leads to a standard error of 0.28. Lastly, using all coefficients and mean domain privacy levels as an input leads to a standard error of 0.27, which is again equal to that of the MGR method; using only the context-factor coefficients of the social, location, or mobile app domain leads to a standard error of 0.28.

Mean-based context-aware regression analysis (MCR)

In contrast to the MGR and CGR methods, which have the goal to predict *mean domain privacy levels*, the MCR and CCR methods do a regression on the fine-grained *context-factor-based privacy levels*. We followed the same approach as in the exploratory study, but this time using only the mean domain privacy levels that were found viable in the exploratory study. **As the number of context-based privacy levels is very high, we report only the mean standard error, as well as the minimum and maximum standard error for every domain and context factor for the MCR and CCR method**. The results can be seen in Table 7.6.

For the location sharing domain, the standard error is similar for both the occasion and requestor context factor instances, whereas the "requestor" privacy levels can be predicted slightly better using all other mean domain privacy levels ($stderr_{avg} = 1.57$) compared to the "occasion" context factor ($stderr_{avg} = 1.53$). When using only single mean domain privacy levels, the ones from the cluster partner (social media domain; see Section 7.2.2) lead to the smallest standard errors ($stderr_{avg} = 1.58$ for "occasion"; $stderr_{avg} = 1.54$ for "requestor"). The same holds for the social media domain, where the best domain for predicting the context-based privacy levels is the location sharing domain ($stderr_{avg} = 0.80$ for "topic"; $stderr_{avg} = 0.72$ for "recipients"). Adding the other mean domain privacy levels does not increase the precision. However, in contrast to the prediction for the location sharing domain, the standard errors using the shopping mean domain privacy level are only slightly higher ($stderr_{avg} = 0.81$ for "topic"; $stderr_{avg} = 0.73$ for "recipients"). For the privacy levels for the different app categories in the mobile app domain, all domains lead to the same precision ($stderr_{avg} = 0.43$). Using all of them together again slightly reduces the standard error ($stderr_{avg} = 0.72$). Finally, the shopping domain can be predicted best by both the social media and mobile phone mean domain privacy levels ($stderr_{avg} = 0.41$ for "data type"; $stderr_{avg} = 0.94$ for "stakeholder"). Using all mean domain privacy levels does not decrease the standard error – neither for the "data type", nor for the "stakeholder" context factor.

Context-factor-based context-aware regression analysis (CCR)

As described in the results section of the exploratory study, we use the context-based privacy levels with $p < 0.1$ for the validation study of the CCR analysis. The standard errors for the different domains are shown in Table 7.6 together with the results of the MCR analysis. For each target domain, we calculated the average, minimum, and maximum standard error for the different context-factor-based privacy levels, using either the coefficients from all other domains, or only from one of the three other domains.

TABLE 7.6: Validation study results for the MCR and CCR analysis.

Target domain	coefficients	MCR stderr			CCR stderr		
		avg	min	max	avg	min	max
Location - occasion	all	1.57	1.27	1.73	1.50	1.22	1.70
	mobile	1.65	1.31	1.86	1.67	1.32	1.87
	social	1.58	1.28	1.74	1.52	1.25	1.74
	shopping	1.65	1.33	1.85	1.64	1.33	1.86
Location - requestor	all	1.53	1.43	1.59	1.49	1.39	1.57
	mobile	1.62	1.51	1.74	1.61	1.50	1.73
	social	1.54	1.43	1.59	1.48	1.39	1.58
	shopping	1.61	1.50	1.70	1.61	1.50	1.70
Social - topic	all	0.80	0.66	1.09	0.76	0.63	1.09
	mobile	0.83	0.69	1.11	0.82	0.69	1.11
	location	0.80	0.66	1.09	0.77	0.65	1.07
	shopping	0.81	0.67	1.08	0.81	0.66	1.08
Social - recipients	all	0.72	0.58	0.90	0.67	0.57	0.81
	mobile	0.75	0.62	0.96	0.75	0.61	0.96
	location	0.72	0.58	0.90	0.70	0.57	0.84
	shopping	0.73	0.60	0.88	0.72	0.60	0.88
Mobile - category	all	0.45	0.42	0.49	0.45	0.42	0.50
	social	0.47	0.43	0.50	0.46	0.43	0.50
	location	0.46	0.43	0.50	0.45	0.42	0.50
	shopping	0.46	0.43	0.49	0.46	0.43	0.49
Shopping - data type	all	0.46	0.41	0.50	0.44	0.38	0.50
	social	0.46	0.41	0.49	0.45	0.41	0.49
	location	0.46	0.43	0.50	0.46	0.41	0.50
	mobile	0.46	0.41	0.50	0.45	0.39	0.49
Shopping - stakeholder	all	0.94	0.85	1.10	0.91	0.82	1.09
	social	0.94	0.85	1.09	0.94	0.85	1.08
	location	0.96	0.89	1.09	0.96	0.86	1.09
	mobile	0.94	0.85	1.10	0.95	0.84	1.10

For the "occasion" privacy levels in the location sharing domain, the CCR method produces better results for all input domains, except for the mobile app domain. In the following, we will compare the standard errors of the MCR method with the standard errors of the CCR method. The difference in the standard error will be denoted by Δ and is calculated as $\Delta = stderr_{MCR} - stderr_{CCR}$. Especially when using all viable coefficients, the standard error can be reduced to 1.50, which is a difference of $\Delta = -0.07$ compared to the MCR method. Using only social media coefficients allows us to reduce the stderr by 0.06 to a final value of 1.52 for the CCR. Using the context-based privacy levels from the intelligent shopping domain can only improve the stderr by 0.01 compared to the MCR analysis. Finally, the stderr using the coefficients from the mobile app domain increases by 0.02 to 1.67, which is the highest standard error of all input combinations for the location sharing domain. The CCR produces better results for the "requestor" context factor of the location sharing domain as well: when using only social media input, the stderr decreases to 1.48 ($\Delta = -0.06$); it stays almost the same for the mobile app (stderr = 1.61, $\Delta = -0.01$) and intelligent shopping coefficients (stderr=1.61, $\Delta = 0.00$). Again using all coefficients from all three domains leads to a prediction which is $\Delta = -0.04$ (stderr = 1.49) more precise compared to the MCR method. The social media domain can also profit from the more fine-grained input of the CCR domain: Using all coefficients, we can reduce the stderr by $\Delta = -0.04$ for the "topic", and by $\Delta = -0.05$ for the "recipients" context factor. Also, using only the coefficients from the location sharing domain reduces the standard errors for both "topic" (stderr=0.77, $\Delta = -0.03$) and "recipients" (stderr = 0.70, $\Delta = -0.02$). However, the prediction using only mobile app (stderr = 0.82, $\Delta = -0.01$ for "topic", stderr=0.75, $\Delta = 0.00$ for "recipients") or intelligent shopping (stderr = 0.81, $\Delta = 0.00$ for "topic", stderr=0.72, $\Delta = -0.01$ for "recipients") coefficients improves the precision only slightly. In the intelligent shopping domain, only the precision using all context-based privacy levels improved the regression precision (stderr = 0.91, $\Delta = -0.03$ for "stakeholder", stderr = 0.44, $\Delta = -0.02$ for "data type"). The precision using only coefficients from either the social media (stderr = 0.94, $\Delta = 0.00$ for "stakeholder", stderr = 0.45, $\Delta = -0.01$ for "data type"), location sharing (stderr = 0.96, $\Delta = 0.00$ for "stakeholder", stderr = 0.46, $\Delta = 0.00$ for "data type") or mobile app (stderr = 0.95, $\Delta = +0.01$ for "stakeholder", stderr = 0.45, $\Delta = -0.01$ for "data type") domain did not change much. Finally, the mobile app domain also could not profit significantly from the increased granularity of the CCR method. Using coefficients only from the social media (stderr = 0.46, $\Delta = -0.01$) or location sharing (stderr = 0.45, $\Delta = -0.01$) domain only slightly decreases the standard error, whereas it stays the same for the intelligent shopping domain (stderr = 0.46, $\Delta = 0.00$) and also when using all domains together as an input (stderr = 0.45, $\Delta = 0.00$).

7.4 Discussion

7.4.1 Predicting mean domain privacy levels using MGR vs. CGR

We presented two different approaches for predicting the mean domain privacy level (the mean privacy level computed over all privacy levels of a domain) of a domain: first the MGR approach that uses the other mean domain privacy levels as an input, and second the CGR method that uses the privacy levels for the different context factor instances (like the privacy level given for social network posts about food, or a location sharing privacy level that has to be applied when a family member requests the location) in addition to the mean-based privacy level. Usually, one would think that more data leads to a higher precision (e.g. a lower standard error). However, this is only the case for the location sharing domain, where the CGR method leads to lower standard errors when using the social media or shopping privacy levels, and especially when all domain data can be used. Still, the size of the effect is relatively small, with a standard error improvement ranging between 0.03 for the intelligent shopping domain to 0.06 (or 5%) when using all context-based domain privacy levels as an input, resulting in a final standard error of 1.20. For all other domains, except for the shopping domain, where the CGR method produces a slightly better result (standard error improvement of 0.01) when using location sharing privacy levels, the CGR method offers exactly the same precision as the simpler MGR approach.

Taking a look at the coefficients used for the location sharing prediction (Table 7.3), we can see that there are two context factor instances which are of major importance from the social media domain: the "topic" context factor instances "movies" and especially "events". Those two post topics are occasions in which users typically also share their location (especially for events), which might lead to their suitability for a prediction, which then leads to a decreased standard error when added to the set of coefficients. The same seems to hold for the amount of items bought, which is used for predicting the location sharing level when only shopping privacy levels are available. People like to share their location during shopping either when they have bought expensive products, or when they have bought an extraordinarily high amount of items, for example at a sale or at a factory outlet store. To conclude, we can state that the simple MGR approach works very well for most of the domains. There are only some domains where the increased data set of the CGR method can improve the precision, like the location sharing domain. Which other domains are also suitable for CGR should be a research topic of future work. As a rule of thumb, it seems like CGR can profit from its context-based privacy levels from the other domains, if some of them are very similar or are often used together with the privacy levels of the target domain.

7.4.2 Predicting context-based privacy levels using MCR vs. CCR

When it comes to predicting the context-based privacy levels, i.e. the different privacy levels depending on the context factors mentioned in Chapters 4 to 6, the simplistic mean domain privacy level-based approach (MCR) still performs well. However, in this case, the context-based method (CCR) can outperform the MCR in most cases. In the location sharing domain, where the CGR already outperformed the MGR, the CCR leads to a standard error that is on average 2.6%–4.4% more precise than the MCR approach when using all other domains as an input. The improvement is even larger for the social media domain, where the standard error is reduced

by 5%–7% when using all input data. For the shopping domain, the improvement amounts to 3.2%–4.3%. The mobile app category remains the only one where the CCR approach led to only meaningless improvements. In the domain of mobile app permissions, the context-based privacy levels were based on the app category as a context factor. However, none of the other domains have a similar context factor, so the other context-based privacy levels, which were based on the requestor or the occasion, for example, were not of any help, so that the CGR approach could not lead to an improvement of the prediction precision. We therefore speculate that the performance of the CCR depends on the semantic distance between the context factors of the target domain and the input domains. Whether this assumption can be generalized should be investigated in future work. In general, we conclude that, for predicting context-based privacy settings, the context-based CCR method should be preferred for most cases, if the data is available. But if the context factors do not match well, e.g. the semantic distance between them is high, the CCR seems not to lead to any advantage. However, the simplistic MCR approach that uses only the mean domain privacy levels from the other domains performs surprisingly well, even for predicting fine-grained privacy levels.

7.4.3 Which data set is to be used for a prediction?

After deciding on a suitable prediction method (either mean-based or context-based), the next question is which data should be used for the prediction, or whether the available data is sufficient for a prediction. In general, if data from all other domains is available, this data should also be used. In our experiments, we did not identify any case where the precision decreased when using all domain data instead of only a specific domain. If privacy levels of only some of the domains are available, or if the data has to be acquired/processed first, it is best to think in clusters. In the experiments, we identified two domain clusters, within which each domain is particularly suitable for predicting the other domain. The first is the location-sharing/social media cluster; the second is the mobile app permission/intelligent shopping cluster. Whenever the privacy levels from the cluster partner are available or can be acquired, they should be preferred before those of all other domains. If data from other domains is already available, it should be added as well, although it will not increase the precision very much. We recommend not to add further domain data if the additional data must be acquired first, and the acquisition would lead to an increased user burden or an excessive computing overhead.

7.4.4 The privacy paradox in privacy recommender systems

The goal of our research is to help users to tune their privacy settings so that they disclose as little private information as needed while still keeping the services (for example the social network or smartphone) usable. However, in order to allow an automatic prediction of privacy settings, we in fact *need additional information from the user* as an input for a meaningful prediction. This fact, known as the "recommender systems privacy paradox", has been a subject of research for several years [322]. There are several approaches that allow the user to increase her privacy in such a recommender system, for example by using k-anonymity (aggregating the personal information together with 2, 3, 4 or n other data sets) at the cost of prediction precision [322]. Data expiration and data morphing are further methods that can enhance privacy at the cost of the recommendation quality [322]. Other approaches try to adapt the recommender system itself to be more privacy-aware, for example

by employing a differential privacy mechanism in matrix factorization approaches [121]. Another PLA-based framework selects a personalisation method at runtime that fits the user's privacy requirements, to enhance privacy while keeping the prediction quality at a similar level [332]. In our studies, we were focused on finding out whether and how well privacy recommendations work in an *optimal* case, where the user's personal information is fully available. However, in future work, we would like to inspect how well our approaches work when privacy-enhancing techniques are included.

7.5 Conclusion

User often neglect their privacy settings, as they often do not see the potential risks that come with oversharing the data. There exist many solutions that try to aid the user in choosing the privacy settings using machine learning, either by using other privacy settings from the same domain as an input, or by utilizing the user's personality and privacy attitudes for a personalized recommendation. However, as this information is not always available, we examined whether privacy settings from other domains can also be used as an input for the prediction. We observed the prediction of a mean domain privacy level that gives only one general user-specific privacy level for a domain as an orientation, as well as the prediction of fine-grained context-based privacy levels that give a distinct personalized privacy level for each combination of context factors. The results show that both types of privacy levels can be predicted already using only the mean domain privacy levels from the other domains. However, the fine-grained context-based privacy levels and the mean domain privacy levels from the location sharing domain can be predicted better using the context-based privacy levels as an input. Although we verified the selected regression coefficients within a validation study, and although we achieved a small increase in the prediction precision using the CGR and CCR method compared to the MGR and MCR method in some cases, we would like to test the suitability of our prediction in future work in an in-the-wild study, where the privacy levels are predicted from actual privacy settings of the users, and check how well the implementation of the predicted privacy levels in the different domains fits the actual desired privacy settings of the user.

In the last chapters, we have seen that individual factors can improve the prediction accuracy in several domains; we examined four different domains as an example. Using machine learning, the individual measures can even be derived automatically without any user burden. However, those algorithms are not always correct; there is always a small chance of an incorrect prediction, no matter how good the algorithm is. Therefore, in our opinion, such a recommender system must always be accompagnied with a privacy UI that helps the user to get an overview of their privacy settings and possible errors, and that also gives the user the possibility to correct these errors easily. The next chapter will describe in more detail which tasks have to be conducted typically when a user has to select her privacy settings, and gives examples for how a user interface could support the user when doing these tasks.

Chapter 8

Motivating and assisting users to reflect their privacy

As mentioned earlier, the concept of our privacy framework is based on two pillars (see Chapter 1): first, offering the user a set of privacy settings that is tailored towards her personality and privacy attitude, and second, a user interface that enables the user to review and adapt these proposed settings. In the previous chapters, we proposed different mechanisms that allow us to propose privacy settings, either based on context and individual factors, or based on other privacy settings of the user. In this chapter, we will discuss the user interfaces needed to complete the concept of the privacy framework. First, we will discuss options for helping users to do friend grouping, which is an indispensable requirement for many privacy recommenders [246]. Afterwards, we will show two user interfaces that cover two different aspects of privacy. The first user interface is targeted towards defining the *audience* (i.e. the recipients) for a sensitive data item. The system, called OmniWedges, allows the user to select the correct audience for a social network post based on the user's friend groups and the tie strength between the user and his friends using a radar metaphor. In contrast, the second of the two user interfaces has the goal to allow the user to define the *granularity* of the private information that should be shared, considering the example of the intelligent retail domain. URetail is tailored especially towards detecting unusual and possibly incorrect privacy settings at one glance, so the user knows where to focus when reviewing the privacy settings. Although both user interfaces are demonstrated with the example of one domain, the approaches can also be transferred to other domains where private data has to be shared (such as, for example, the four domains discussed in Chapters 4, 5 and 6), as we will discuss in this chapter. The chapter ends with a discussion of how in-situ feedback on mobile devices can be employed to allow the user to notify the system about privacy invasions as soon as they are noticed, which consequences users expect from the system when giving feedback, and how this kind of feedback can be used to refine the privacy settings.

8.1 Motivating users in friend grouping

As we have highlighted in earlier chapters, users often share data online with a larger audience than intended. Social networks such as Facebook or Google+ allow their users to create custom friend lists and share content exclusively with these lists. In the last two chapters, we presented approaches that recommend privacy settings for a user based on the user's individual measures for each group of recipients. However, users rarely create friend lists [230]. Known causes for this behavior are the mental effort to group people [336] and usability problems, e.g. regarding the

mechanics, general workflow problems or simply the user interface design [179, 346, 184].

Compared to other sorting tasks, the task of social network friend grouping includes several special challenges to be solved: First, we have seen in the last chapters that privacy preferences are highly individual in several domains like social network posts and location sharing, mobile app permissions or shopping scenarios. Every user has her own different criteria to build groups and to categorize her friends into them, depending on her personal preferences, her personality and privacy attitudes, and also her posting preferences, regarding post topics and intimacy of the shared information. Second, there is no definite answer to the correct assignment of a friend. Some of the friends might fit into multiple groups; some might not fit in any group and will remain unassigned. This leads to the fact that for some friends, it is immediately clear to which groups they should be assigned, whereas the user needs a longer time to think about a correct assignment for other cases, as our study results will show.

Research has tried to tackle this problem by creating new design concepts with an increased usability in order to reduce the mental effort to perform the friend sorting task. Some of the approaches use graph-based interfaces [85, 224], where groups are represented by vertices with the corresponding friends attached as their leaves; others rely on a conventional list-based design [210] improved by an auto-grouping algorithm based on community detection [40]. Nevertheless, the usage of virtual reality to enhance the usability of social network friend sorting on the one hand, and to make the task more interesting and enjoyable by enhancing the user experience on the other hand, has not been discussed in research so far to the best of our knowledge. To be more precise, we try to solve the following research questions:

1. Can we enhance the usability and user experience of the social network friend sorting task using a VR environment and metaphors?
2. Do users prefer a playful approach or an approach that is optimized towards usability ("pragmatic design") for VR sorting?
3. How do the VR designs affect the errors made during friend sorting?

For this purpose, we created two different UI designs. The first one, later called "pragmatic design", is focused on further increasing the usability (in a virtual reality environment) by adapting traditional concepts such as card sorting. The second is geared towards making the sorting task as fun and enjoyable as possible by packaging the task as an interactive VR game ("playful condition"). In a study comparing these to a conventional sorting interface from the Facebook social network website, we found that we could further increase the usability with the pragmatic design. The playful condition was perceived as highly motivating and achieved a significantly higher user experience score, at the cost of an increased error rate. The work presented here is based on already published research [266].

8.1.1 FriendGroupVR Designs

We implemented *two* different VR design approaches to sort and organize social network friend lists, targeting different objectives: the first "pragmatic" approach is optimized towards *usability* in terms of efficiency and performance, whereas the second "playful" approach is focused on maing the sorting task as enjoyable and interesting as possible. Each world was implemented in Unity using an HTC Vive

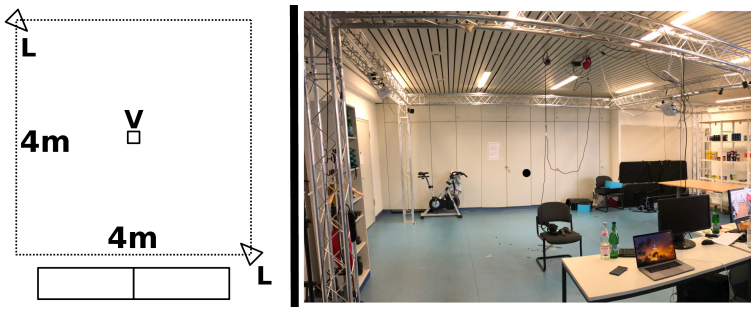


FIGURE 8.1: VR setup in our lab using the HTC Vive. "L" denotes the positions of the lighthouse position trackers, "V" the initial position of the user wearing the HTC Vive

VR Kit. The setup contained a 4m x 4m floor equipped with an HTC Vive Lighthouse setup that allows tracking the user's movements inside the area (see Figure 8.1). Each user movement was reflected in the VR world as well. In order to track hand movements, each user was given two Vive controllers, one for each hand. Grabbing gestures were realized by usage of the trigger buttons of the controllers. For our lab study, we recorded the created friend lists and the contained friend lists locally instead of applying the changes to the user's social network account.

A special problem of the friend sorting task is that the time needed for the assignment of a friend to one or multiple groups can be highly variable. For some of the friends, it is instantly clear to which social circle(s) or which friend list(s) they belong, but for others it is less clear, so that the user might need a few seconds to think before he can conduct the actual assignment task. Therefore, we put a special emphasis on the possibility to interrupt the sorting task between two friends, so that the user has the possibility to think about the best assignment options in advance.

Pragmatic design

According to related work, the metaphor of card sorting is one of the most efficient methodologies [61]; we therefore decided to transfer the open card sorting metaphor into a VR world, leading us to an office metaphor as shown in Figure 8.2. The "cards", i.e. the social network friends, are represented as picture frames ("friend frames") standing inside a bookshelf. Each friend frame consists of the friend's profile picture and first name on the front, and the first and last name on the back, forming a combination of a card sorting and picture sorting metaphor ("card sorting+"), as described in the related work section. Friends can be displayed with ascending tie strength (equivalent to the Facebook friend list order) or sorted by first or last name. As space is limited, the shelf always contains only nine friend frames at a time. To access the other frames, we placed two buttons at the left and right edges of the shelf, allowing the user to access friends that appear earlier or later in the sorted list, respectively (see Figure 8.2).

According to the *stacked cards metaphor*, friend lists are represented as labeled boxes ("list box") in which the user can drag & drop friend frames using a VR controller. As a starting point, the VR world contains the five most frequently used friend lists according to Chapter 4.1, namely "family", "acquaintances", "close friends", "work" and "sport", as a box. Boxes have no physical weight in our VR world, and can therefore be placed in mid-air at any desired location. As the task of arranging friend lists is highly individual, we opted for an "open card sorting" design allowing

FIGURE 8.2: Bookshelf in the pragmatic design, including friends represented by "friend frames" (left) and spawner to create new friend list boxes (right).

users to create arbitrary additional friend lists. To create a new list box, we added the "box spawner" into the environment (Figure 8.2): To create a list box, the user has to touch the red button with the VR controller, which opens a VR keyboard to enter the list name. Pressing the enter button hides the keyboard and spawns the newly created list box, as seen in the figure.

To manage the friend lists, a user typically starts with creating and arranging the list boxes around the shelf. After that, the friend frames are traversed one after another and placed inside one or multiple list boxes that should contain the friend. As soon as a friend frame is placed inside a box, the frame is shrunk to half of its size to save space. If a friend is placed inside the wrong box or if the user decides to assign them to a different list box, he can always empty the box on the floor or grab a picture inside the box and put it into another.

Playful design

In contrast to the former design, here we concentrated on making the sorting task as enjoyable and interesting as possible. We therefore decided to design the approach as an interactive VR game that challenges the user, including gamification elements like high score tables, upgrades and bonus items that should motivate the user in carrying out the task and competing with others. As stated in the beginning of the section, the time needed for finding an optimal assignment is very different from friend to friend. We therefore need a game design which can be interrupted or delayed at certain points in time to allow the user to take her time for the assignment decision. As related work has shown, most of the friends (about 90%) are assigned only to one friend group; we therefore decided on a game with a linear action line, where only one friend is part of the game at a time, with the possibility to manually go back to a friend again if he or she has to be added to multiple friend groups. We came up with the idea of a "can knockdown" game, where the user can assign her friend to friend lists by shooting dispatched "friend balls" to different can stacks representing the available friend lists. Using this design, the user can always wait and think about the correct assignment, before she starts the dispatch of the friend ball.

In a typical workflow, the user first creates the needed friend lists using a tool similar to the box spawner in the pragmatic design. After this task is finished, the friend lists are represented by can stacks ("list stacks") at a distance of about five meters in front of the user. The user then starts the assignment phase, where a ball

FIGURE 8.3: Can knockdown game in the playful design.

representing each social network friend is dispatched in the direction of the user one after another. The player uses a bat to redirect the friend ball to a can stack corresponding to the friend list the user should be assigned to. As mentioned before, each friend ball is dispatched only once. If the user wants to add a friend to multiple groups, he has to press the "back" button on the control panel (see below) to display the last friend ball again and add her to another group. Depending on how many cans the user is able to hit with the friend ball, the user gains points to be added to his personal high score.

A screenshot of the playful VR world from the user's initial position can be found in Figure 8.3: the shelf on the left side of the user (Figure 8.4 left) is used to create and arrange the friend lists, similar to the pragmatic design. In the shelf, friend lists are represented by a small board with the list name written on the front. Similar to the other design, the five most frequently used friend lists are already created in advance and placed at the bottom of the shelf. If the user wants to create a new friend list, he touches the button, which opens a keyboard to enter the friend list name, exactly like in the pragmatic design. To use a friend list in the can knockdown game, the user has to place a friend list board in one of the containers in the shelf, which will display a can stack in the game at the respective location (e.g. if the board is placed in the container to the left of center, the corresponding can stack will also be shown to the left of center in the game). At the right hand side of the user is a control panel (Figure 8.4 top right) which allows the user to switch forward or backward between the social network friends, and a button to start and pause the game at any given time, for example if more time is needed to contemplate the correct friend list assignment. Using the control panel, the user can also go back to an earlier friend and dispatch her friend ball another time to assign her to another friend list.

At the front, the user is facing a panel (Figure 8.4 bottom right) which displays the name and profile picture of the next friend to be sorted, together with the current score and the remaining time for the currently collected bonus item (see below). When the game is started by pressing the "start" button, the user has five seconds to think about the correct list stack that he wants to aim at. The five dots at the top of the panel represent the time in seconds that is remaining. When the last dot turns from gray to red, the friend ball is dispatched towards the user. If no can stack is hit, the same friend ball is again dispatched for another try. When the user does

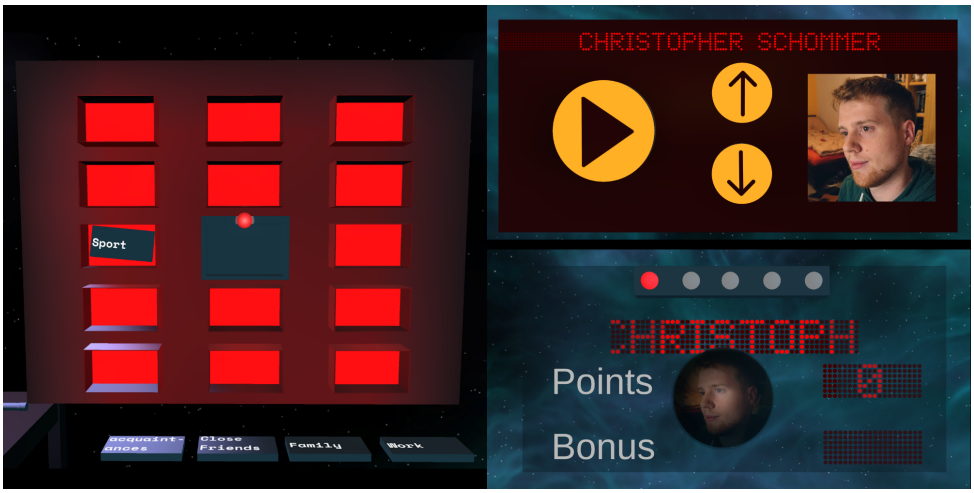


FIGURE 8.4: Friend list management shelf (left), score and dispatch panel opposite the user, dispatching a friend ball (bottom right) and control panel to switch the current friend and start/pause the game (top right) in the playful design.

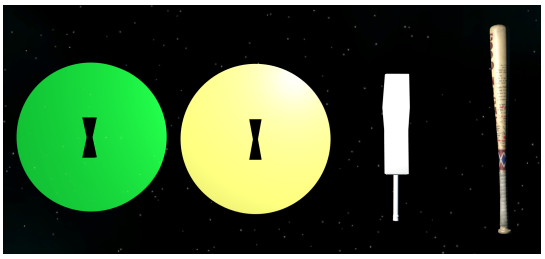


FIGURE 8.5: Different bats available to the user with ascending difficulty from left to right: Round bat with magnifier functionality, round bat without magnifier functionality, cricket bat, baseball bat.

not want to assign the friend to any group, he can aim for the monster at the upper left of the VR world, which will then eat the friend ball, so that it is not dispatched again. If the user hits the wrong can stack, he can always undo the last assignment by hitting the "undo" buzzer directly in front of him. To further motivate the user, we integrated "bonus balls" into the game, which are dispatched in the direction of the user at randomized times. Collecting each bonus ball activates a special upgrade for a limited time, for example a score multiplier, or an increase of the friend ball size.

The user has a choice of different bats (Figure 8.5) with different difficulties: The easiest bat is largest and catches the ball so that the user has the possibility to aim and shoot at the desired location by pressing a button. The second easiest bat has the same size, but directly deflects the ball without catching it first. The remaining two bats have the same behavior with a smaller size, making it more difficult to hit the ball. The more difficult a bat is, the more points are rewarded for each can hit. When all friend balls have been processed, the game stops and the user's high score is displayed on the high score table in the upper right of the VR world, along with the high scores of other users, and the friend lists are stored.

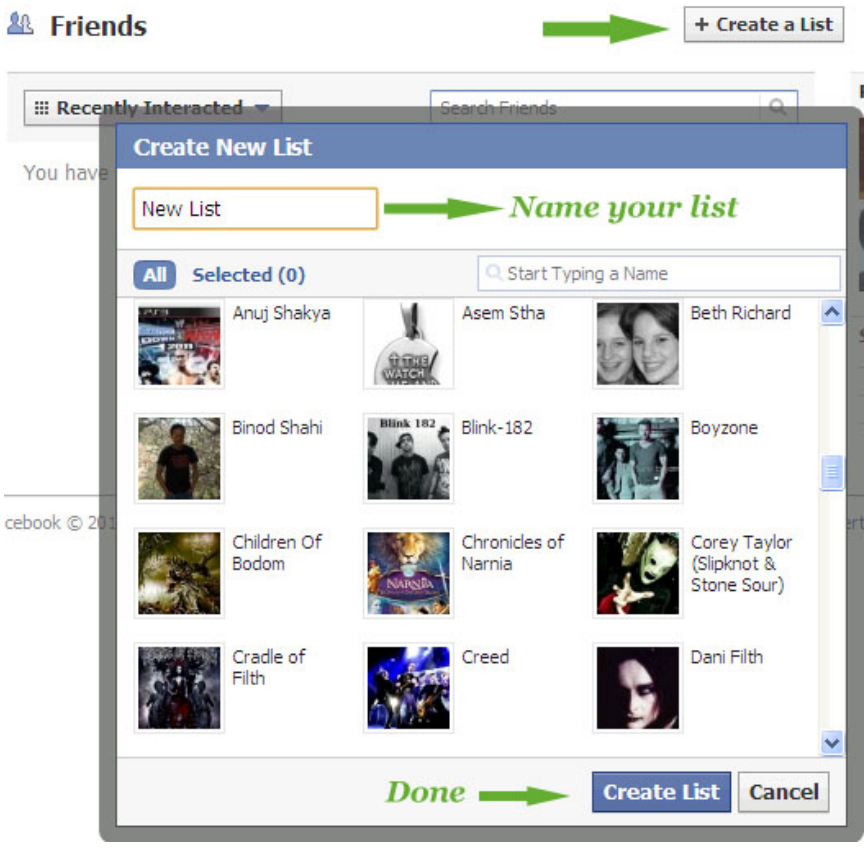


FIGURE 8.6: Friend group creation interface from Facebook. *Image source: https://www.jucktion.com/tech/how-to-be-invisible-on-facebook-chat/*

8.1.2 User study

We had the goal to find interaction designs using VR for the creation and maintenance of social network friend lists that would be both more efficient and also more enjoyable than the current standard. As a reference interface, we used the Facebook interface for creating friend groups as shown in Figure 8.6. In the Facebook interface, the user is shown all friends in a grid view, and has to select the friends to be added to the list by clicking on them.

In order to measure the differences from the Facebook UI, we conducted a lab study at our department, where the participants had to use both VR designs as well as a standard interface from the Facebook social network site using a desktop PC as a baseline. With each interface, the participants had to assign their 40 closest friends (according to the Facebook friend ordering) to friend groups. For each condition, we recorded usability and user experience scores using the AttrakDiff [150] questionnaire as well as an error rate (for example friends missing from a group, or friends assigned to the wrong group), as desribed below in more detail. To reduce training effects and to get users used to the VR environment, we implemented another VR "training" world which shows the user an overview of the 40 friends that have to be assigned in the experiment (Figure 8.7). To further reduce training effects, the order of conditions was permuted for each participant so that each sequence of conditions appears equally often during the study, leading to $3! = 6$ different orders.

The procedure was the same for each participant but with a different order of

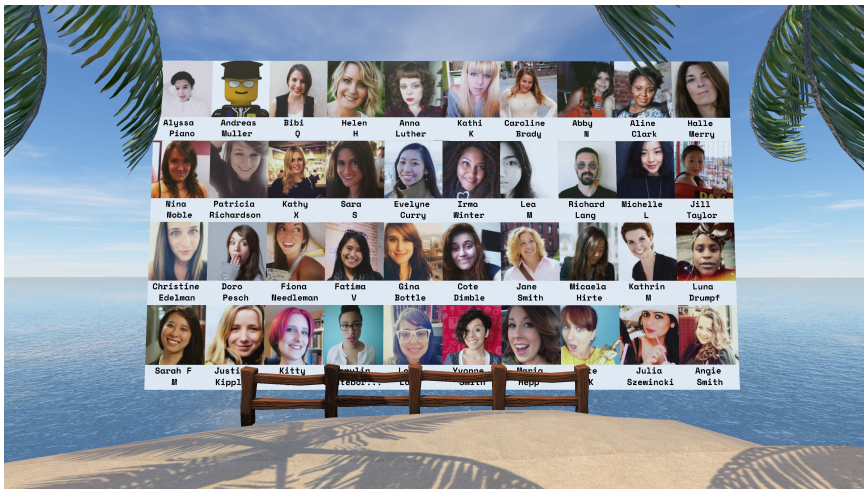


FIGURE 8.7: Training world with 40 friends that the user is shown before the experiment starts.

conditions, as stated before. After signing a consent form and agreeing to the privacy policy, the participant had to fill in a questionnaire about demographic data and previous experience using the Facebook friend grouping tool and virtual reality setups. She was then given a desktop screen to enter her Facebook login data. With the aid of the Selenium web browser automation toolkit[1], a Python script then traversed the participant's friend list and extracted the friend names and profile pictures for later use during the study. After the process was finished, the participant was given instruction in the VR hardware, and had to put on the headset for the first time. We started the training level and gave her the time to get familiar with the VR world and the controllers, and to have a first look at the friends to be sorted and to contemplate the friend lists and the assignments to be made. When the participant stated she was ready, the training world was closed, and the main experiment phase started.

In the main phase, the three interface conditions were tested one after another in a different order, as stated above. For each condition, the participant was given an introduction to the interface with all of its interaction possibilities and some time to get familiar with it and to test each functionality once. When she stated she was ready, the world was reset, and the participant had to do the friend grouping with her 40 friends until she stated she was finished. Participants were told that they should do the task seriously, as wrong assignments would be recorded. In the following, the participant had to fill in several questionnaires about the current condition: the AttrakDiff questionnaire [150] measuring usability (PQ) and user experience (HQ-I, HQ-S), the NASA TLX capturing the mental and physical workload, and an MSAQ questionnaire asking about motion sickness in the VR conditions, as well as a custom questionnaire asking whether the interface was motivating or fun to use, and whether the participant thought it could be integrated into her daily life, on a five-point Likert scale. After a five-minute break to rest and recover, this procedure was repeated for the two other conditions. For each condition, we recorded the overall time spent on soritng. At the end of the study, the participant was asked which was their favorite interface, and had to traverse the friend lists created to check for errors made during the assignment. We recorded the following error measures:

[1] https://docs.seleniumhq.org/ (last accessed: 2020-03-09)

measure	M_{FB}	$M_{playful}$	$M_{pragmatic}$
PQ	-0.15	0.61	1.99
HQ-S	-1.81	2.02	1.13
HQ-I	-0.54	0.96	1.21
FUN	1.57	4.77	4.57
MOTIV	1.65	4.23	4.33
DAILY	2.10	2.87	3.77
Workload	30.61	39.42	22.86
Errors	8.50	11.93	7.60
Time(s)	418	587	512

TABLE 8.1: Results for the usability and user experience scores including pragmatic quality (PQ), hedonic quality regarding stimulation (HQ-S) and identification (HQ-I) and the custom questions asking about fun (FUN) and motivation (MOTIV) to do the task and suitability for everyday usage (DAILY), as well as the Nasa TLX workload, time spent on sorting and the error rate.

- Person missing from a group (MISS)
- Person added despite not belonging to the group (TOOMUCH)
- Wrong group label (LABEL)
- Group should be split into multiple groups (SPLIT)
- Multiple groups should be merged into one group (MERGE)

8.1.3 Results

In total, we had 30 participants in the study, 18 female and 12 male. Participants were recruited at our university using postings and the university's social network group. As a compensation, a 25€ Amazon voucher was raffled off among all participants. Ages ranged from 19 to 50 (mean=26.67, SD=5.474), representing a good portion of typical social network users [2]. When asked about their experiences with virtual reality, 12 people had no experience (40%) and 5 almost no experience (16.7%). 25 people answered that they had never used Facebook's grouping interface (83.3%), while 5 people had used it (16.7%). On average, the main experiment was completed within 64 minutes.

The experiment results can be found in Table 8.1. Depending on whether an F-test showed a normal distribution of the data, we performed pairwise paired T-tests or Wilcoxon signed-rank tests to compare the pragmatic quality (PQ), also known as usability; the measures from the custom questionnaire asking about experienced fun (FUN) and motivation (MOTIV) and suitability for daily use (DAILY), the NASA-TLX workload values; times needed for sorting; the error rates and the hedonic scores HQ-I and HQ-S measuring the user experience between the three conditions.

The usability (PQ) was highest for the pragmatic interface ($M = 1.99$) and significantly better than for the playful interface ($M = 0.61, T = 4.6, p < 0.001$) which is itself significantly more usable than the Facebook standard ($M = -0.15, T = 3.1, p = 0.004$). Regarding the user experience, the user could identify significantly better with the pragmatic interface ($M = 1.20$) than with the playful interface ($M =$

[2] https://www.statista.com/statistics/274829/age-distribution-of-active-social-media-users-worldwide-by-platform/ (last accessed: 2020-03-09)

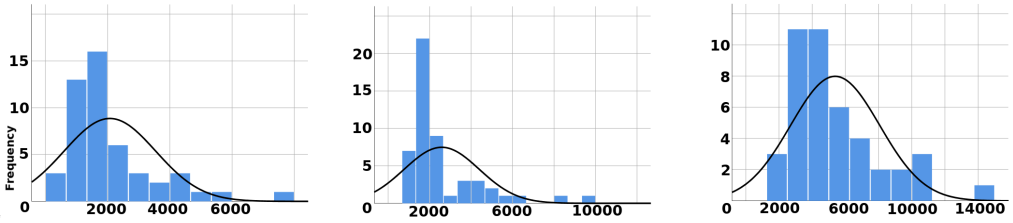


FIGURE 8.8: Distribution of the time needed to assign a friend for three representative subjects.

0.0.96, $T = 2.13, p = 0.042$) which was again better than Facebook ($M = -0.53, T = 8.38, p < 0.001$), but felt most stimulated by the playful interface ($M = 2.04$), followed by the pragmatic VR design ($M = 1.13, Z = -4.50, p < 0.001$) and distantly followed by the Facebook UI with a significantly lower score ($M = -1.81, T = 14.471, p < 0.001$). The FUN was on average also highest using the playful design ($M = 4.77$) although we could not prove the difference from the pragmatic interface to be significant ($M = 4.57, Z = 0.965, p = 0.334$). The Facebook interface was again rated significantly worse than the pragmatic interface ($M = 1.57, T = 4.79, p < 0.001$). The most motivating interface is the pragmatic interface ($M = 4.33$) according to the mean values, but is again not significanlty better than the playful interface ($M = 4.23, Z = 1.62, p = 0.09$). The Facebook UI is again significantly worse than the playful UI ($M = 1.65, Z = 4.75, p < 0.001$). The same order holds for the suitability of integrating the UI into everyday social network usage: the pragmatic interface ($M = 3.77$) significantly outperforms the playful interface ($M = 2.87, Z = 3.24, p = 0.001$), which is again significantly better than the current standard on Facebook ($M = 2.10, Z = 2.39, p = 0.017$). The time in seconds needed to perform the grouping task was lowest with the Facebook interface ($M = 418$) and significantly higher with the pragmatic ($M = 587, T = 2.722, p = 0.011$) and playful VR designs ($M = 588, T = 2.50, p = 0.018$). A visual analysis on the time distributions for the times needed to assign a single friend showed that the times are very different for some of the users, supporting our assumption that an interface is needed that allows users to pause the sorting task, as the time needed for an assignment can differ substantially. Figure 8.8 shows the time distribution for three representative subjects of the study.

The motion sickness (MSAQ) scores, ranging from 11.1 (best) to 100 (worst), were very low for both VR interfaces and did not differ significantly ($M_{pragmatic} = 16.02, M_{playful} = 16.37, Z = 0.991, P = 0.322$), attesting that motion sickness was not a noticeable problem in our UI designs. The pragmatic interface received on average the smallest error rate ($M = 7.60$). Nevertheless, the error rate using the baseline interface is not significantly higher ($M8.50, Z = 0.419, p = 0.675$). The playful interface led to the highest error rates, which are significantly higher than for the standard Facebook interface ($M = 11.933, Z = 2.204, p = 0.027$). The same holds for the workload, which is highest for the playful interface ($M = 39.42$), significantly lower for the Facebook UI ($M = 30.61, T = 2.79, p = 0.0009$), and lowest for the pragmatic UI ($M = 22.85, T = 2.90, p = 0.007$). A detailed overview on the error rates and the different workload items can be found in Tables 8.2 and 8.3. We can clearly see that the main cause for the higher workload in the playful design is that the VR game was perceived as challenging, as the mental demand ($Z = 4.19, p < 0.001$), temporal demand ($T = 4.37, p < 0.001$) and effort ($Z = 4.22, p < 0.001$) are significantly higher compared to the pragmatic interface. The frustration was highest

measure	M_{FB}	$M_{playful}$	$M_{pragmatic}$
MISS	2.5	4.6	1.43
TOOMUCH	0.57	4.6	0.57
LABEL	0.03	0.03	0.1
SPLIT	0.23	0.27	0.17
MERGE	0.07	0.03	0.03

TABLE 8.2: Detailed results for the average number of errors per participant for the different error types.

measure	M_{FB}	$M_{playful}$	$M_{pragmatic}$
Mental demand	29	41.17	22.67
Physical demand	10.50	44.17	35.67
Temporal demand	39.50	47.83	25.67
Performance	30.33	34.83	20.17
Effort	26.33	42	21.50
Frustration	47.67	26.50	11.33

TABLE 8.3: Detailed results of the NASA TLX questionnaire for the three different conditions.

using the Facebook interface, supporting our claim that friend grouping is perceived as a very frustrating and uninteresting task. Using VR, the frustration is significantly lower for both the pragmatic ($T = 6.79, p < 0.001$) as well as the playful design ($T = 3.65, p = 0.001$). The most favored interface was the pragmatic design (73.3%) followed by the playful design (23.33%). Only one participant claimed to like the standard Facebook interface best. We observed different behaviors regarding the choice of bat used throughout the game: 17 participants used the "sticky" bat and 9 used the "deflective" bat most of the time ($> 80\%$ of the time); three switched between the types. One used the baseball bat exclusively. However, we did not find any significant difference between these usage groups for any of our measures.

8.1.4 Discussion

Increased usability using VR

We presented two VR friend grouping interfaces, one geared towards maximizing the usability, and one towards maximizing the fun and motivation when sorting friends. Comparing the usability scores of the interfaces, we can see that both VR interfaces were perceived as significantly more useful than the Facebook interface. The pragmatic design achieved the highest usability score, 1.99, which is very close to the theoretical maximum of 2.50 on a scale from -2.5 to 2.5, indicating that we achieved the goal of increasing the usability compared to the current standard. Interestingly, the interaction time was lowest for the Facebook interface, although it was rated as having a significantly lower usability. On the other hand, the error rate was higher for the Facebook interface, leading to the assumption that the Facebook interface is fast to use on one hand, but is complicated and leads to an increased error rate on the other hand, which leads to a smaller perceived usability of this interface.

Challenging game design leads to increased error rates

The error rate is on average lowest for the pragmatic interface, although the difference to the standard interface is not significant. As stated earlier, we designed the playful design to be challenging for the user, including bonuses, high scores, and different levels of difficulty using different bats. This is also reflected in the perceived workload according to the NASA-TLX, where the mental and physical effort in particular are higher compared to the other interfaces. Nevertheless, the frustration is low compared to the Facebook interface, indicating that the stress was perceived to be positive. However, the challenges may also lead to the increased error rate, which is also highest for the playful interface. The game may have been too challenging, or the design as a game may have led the participants to take the task less seriously and pay less attention to a correct sorting; which of these factors led to the increased error rate should be further investigated in a follow-up study.

Significantly improved user experience, not only for the playful condition

The differences for the user experience scores are again larger than for the usability scores when comparing the VR designs with the standard interface. The playful design received a very high stimulus and FUN score, again indicating that the game was perceived as challenging, stimulating and fun to use. But the pragmatic design, which was not optimized towards user experience, also achieved a high user experience score, which was significantly higher than for the Facebook baseline. The pragmatic design was voted to be most motivating, although not significantly more so than the playful interface. One reason why it was rated as being more motivating on average might be the succesful combination of an appealing and interesting user interface, which still provides a high usability without trying to challenge the user. Which factors led to the higher motivation should therefore be investigated in a follow-up study.

Conclusion on the favored interface

Taking all the aspects into account, the results indicate that VR designs are perceived as more useful on the one hand, and as more fun and motivating on the other, which gives them a clear advantage over the current mouse & keyboard interface. However, such conventional interfaces have the advantage that every computer is equipped with a mouse and keyboard; the audience that can use the Facebook interface is therefore currently significantly larger than those who own a VR setup at home to do the friend sorting with one of the two VR designs. Nevertheless, VR interfaces will gain importance in the next few years, as the number of VR users is increasing exponentially [3].

As stated in the introduction, one of the major problems of friend sorting is that the task imposes a high mental demand, making it a task that is often avoided. A first approach is therefore to present the task as an interesting and challenging game, as in our playful design. However, a playful design has the drawback that it can lead to an increased error rate according to our results. We speculate that this might be caused by the task being taken less seriously, or users being lost in the game without paying attention to the actual task, leading to a decreased quality of the desired results. Therefore, according to our results, the method that is preferred by

[3] https://de.statista.com/statistik/daten/studie/426237/umfrage/prognose-zur-anzahl-der-aktiven-virtual-reality-nutzer-weltweit/ (last accessed: 2020-03-09)

users is a VR design which is targeted towards usability and that could be enhanced with some small game elements, but without losing the focus on the actual task too much. These results confirm the study findings about card sorting, which was already shown to be very efficient using a desktop interface [61], and which seems to be efficient for sorting within a VR world as well. Regarding the differences in time needed for assigning a friend to a list, our results indicate that this time requirement indeed is very diverse, making it important to design a user interface or a VR sorting game so that it can be paused at any time, especially between the items to be sorted. Whether these assumptions can be proven to be true remains for a follow-up experiment, where we will take a closer look at the effects that led to the increased error rates in the playful condition.

8.1.5 Conclusion

Neglecting privacy settings in online social networks can lead to serious harms, but privacy functionalities like friend lists are rarely used in social networks, often due to the fact that the mental effort for creating friend lists prior to their usage is too high, leading users to either censor their posts or to publish more information than they originally intended to. Related work focused on improving the usability of conventional desktop interfaces for friend sorting. In this section, we took a first look at how friend sorting interfaces could look in virtual reality. We proposed one interface focused on usability by taking the idea of card sorting into VR, and a second interface having the goal to maximize the user experience by wrapping the sorting task in a challenging game. A comparative study with the Facebook sorting interface as a baseline has shown that both interfaces achieved their goal of improving the usability and user experience, although the error rate significantly increased within the playful design. However, which distinct factors led to the increased error rate, and which factors led to the increased user experience scores, should be further studied in future research.

8.2 OmniWedges: area-based audience selection for social network posts

Having created the friend groups as described in the last section, the user still has to remember which friends are inside the created friend groups. Currently, the privacy dialogues of social network sites typically offer to share the post with all friends or friends of friends at first sight. There is also an option to go for *custom* privacy settings for a post, opening a new window where the user can enter names of friends and friend groups that should be either included in or excluded from the post. Although this option in theory gives the user the ability to define the visibility for a post differently for every user, she still has to remember which friends are in the friend lists that should be included in or excluded from the post. There is no possibility to see a list of all friends or their profile pictures, or take a look at the members of the friend lists. Additionally, it can be very burdensome to type in all names of users that the user wants to exclude, keeping in mind that a typical Facebook user often has more than 1000 friends. Therefore, users often stick to the default option and share the post with all friends, although the intended audience is actually smaller [139]. Radar interfaces have been used succesfully in research for giving a better overview on privacy settings [69], also further motivating users in adapting them on a regular basis [69].

Nevertheless, the available space inside a radar interface is limited compared to an interface based on a list where the users are listed one by one, which can be extended endlessly using scrollbars. The research community on visualization focuses mainly on mapping-based and clustering-based techniques to reduce the complexity and space needed in a user interface [254]. Nevertheless, those algorithms are usually applied on unstructured data sets like image databases.

In contrast to those approaches, the domain of social network audience selection faces two additional challenges: first, the social network friends cannot be seen as an unstructured database. Each friend has a different role and a different importance to the user, based on factors like the tie strength between the user and the friend in question, or special social ties like family members, neighbors, or friends from a sports club. Earlier work on user interfaces for selecting the social network audience has already shown that aligning the users based on their tie strength to the original poster allows a faster and more intuitive selection of the right post audience, especially for posts with potentially awkward content [180]. Therefore, a visualization that is able to efficiently display and select social network friends needs to take care of each of those individual factors in order to support the user. Second, using clustering-based algorithms like multi-dimensional scaling (MDS) to form a hierarchical structure and to show only one representative of similar images in the user interface is not a solution, as every friend can be seen differently when it comes to privacy decisions, even if the friends are members of a similar social circle, like family members.

In this section, we therefore concentrate on this use-case as an example to investigate:

1. how an interface based on a radar metaphor can be enhanced to be able to display a user's social network audience, while considering the individual roles and social ties of the audience members;

2. whether the error rate (like sharing a post with an unintended audience) can be reduced using such an enhanced radar interface for selecting the post audience; and

3. whether the mental model changes using a radar interface for this task, and which possible effects in usage stratgies arise from this change.

OmniWedges is based on a radar metaphor and aligns the friends of the users based on the tie strength to the user and friend groups they belong to. The audience for a post can be chosen by selecting single or multiple areas inside the radar or subsequently adding single persons. OmniWedges includes functionalities that make it scalable, so it can display *all* of a user's social network friends, even if their number is in the range of several hundreds or thousands, while still giving the user an overview on the most important friends and especially who is selected and who is not. We conducted a study comparing OmniWedges with the standard Facebook interface for custom privacy settings, and were able to show that our approach significantly reduces the amount of errors made during audience selection.

To conclude, our work builds upon a radar design, and introduces several new design innovations and functionalities that make the interface suitable for practical use on Facebook with all Facebook friends. The work presented here is based on already published research [262].

8.2.1 Design of the OmniWedges user interface

OmniWedges allows the user to select the post audience based on the tie strength and the user's friend groups. An example of how the audience can be selected in OmniWedges can be seen in Figure 8.9. Each of the user's social network friends is represented by her profile picture in the user interface, later called the "friend picture". For each friend group that the user created on the social network site, the interface contains a wedge including the friends inside the friend group, represented by their friend pictures.

Inside the whole circle of the UI and also inside each wedge, the friend images are aligned according to their *tie strength* with the user: The closer the friend image is placed to the center, the stronger the tie strength to the user. Currently, OmniWedges is using the tie strength calculation provided by Facebook, i.e. the friend order as displayed on Facebook's friend page, which uses a tie strength calculation for organizing the friends[4]. However, in order to reduce side effects, we let participants review and adapt the friend ordering and the friend lists before the start of the experiment.

In the beginning, no friends are selected. To select a subset of friends, the user drags from the center of a wedge to the outer rim. The area selected by the user is highlighted in a grayish color (see Figure 8.9). The friends inside the selected area of the wedge ("wedge area") are then selected as the recipients of the post. This

[4] https://www.techniquehow.com/2018/07/facebook-order-friend-list.html (last accessed: 2020-03-09)

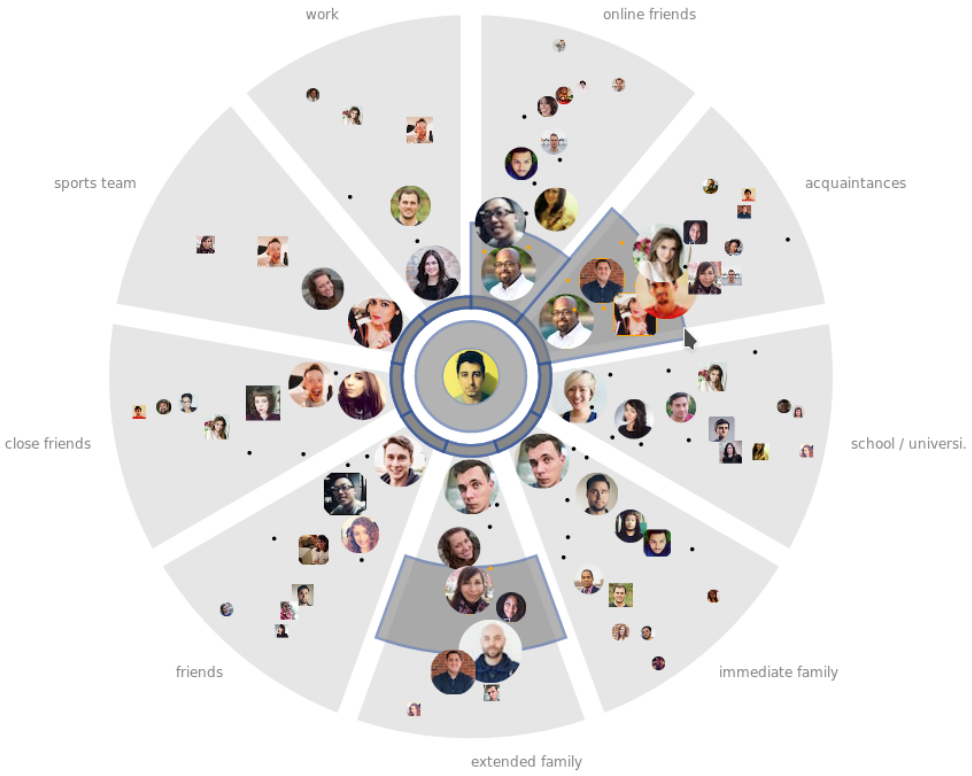


FIGURE 8.9: Radar UI of OmniWedges. The profile pictures shown here and in the remainder of this section are fake profile pictures created by the https://randomuser.me/ API.

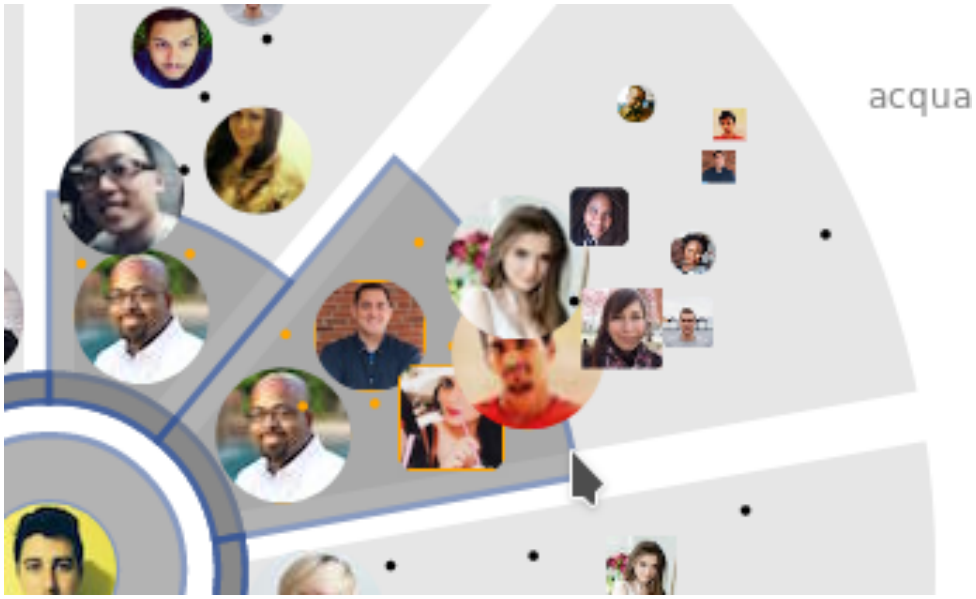


FIGURE 8.10: Detailed view of a wedge during selection process.

procedure can be repeated an infinite number of times. It is also possible to select multiple areas inside the same wedge. The selected friend images are highlighted with a yellow border. Apart from selecting only parts of the wedges, the user can also grab and extend the inner ring in the UI, making it possible to select all friends up to a certain tie strength, independent from the friend group. This procedure can also be repeated as often as desired. Also, single friends can be selected by clicking on the respective friend image.

OmniWedges always adapts the size of the friend images, so that the friend images that are currently of the highest importance also receive the largest amount of space in the UI. In general, the size of the friend image corresponds to the tie strength: whereas closest friends have the highest importance for the user and therefore have the largest image, the size decreases with decreasing tie strength. When the user starts to hover over a friend image or is about to select them, the friend image is magnified and the name of the friend is shown below the enlarged friend image (see Figure 8.10), allowing the user to identify the friends that are currently on the cusp of being selected. The UI again incrementally shrinks the friend image to its initial size and hides the name as the mouse leaves the area.

Below the radar, the UI shows how many friends are currently selected together with a list of the ten closest selected friends. Two buttons in the bottom left corner allow the user to select or deselect all friends.

In order to design and refine the user interface concept, we started with a focus group to discuss the idea. The group was composed of six participants (age: 21-25; mean: 23): five students from different faculties and a process engineer. Four participants were male, two female. All of them used at least Facebook as a social network and one of them was also active on Xing and Instagram. The interview started with general questions about their social network usage behavior, especially use of Facebook friend lists. We continued to discuss possible alternatives for audience selection, introducing a social graph and a sketch of the OmniWedges interfaces in later stages.

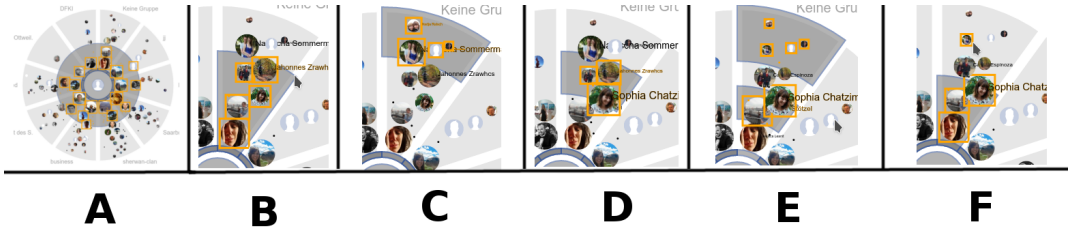


FIGURE 8.11: Functionalities of OmniWedges.

All participants stated that their usage behavior had changed over the last few years. They had posted more content to a broader audience in the past, but more recently, they focused on direct chats and shared less content. Only two out of six participants knew about Facebook friend lists, but they never created or used them for publishing a post. When introducing the sketch of a social graph for Facebook friends, participants stated that such a graph could be used in order to visually select the post recipients by marking areas of the social graph. The proposed concept was perceived as a useful and efficient solution for audience selection. Participants claimed to be willing to use such an interface if it was available on Facebook, and suggested some additional functionalities:

- Exclude a set of friends using the social graph.
- Include/exclude specific friends or a subset of friends inside a friend group.
- Store the selection for reuse in future posts.
- Switch between classic view (the current Facebook standard) and OmniWedges.

We used the input of the focus group to extend our concept and to add additional features (see Figure 8.11), which cover most of the desired functionality mentioned by the participants. In the following, we will describe the functionalities of OmniWedges using typical use-cases in a social network website. The use-cases are depicted in Figure 8.11.

Select all friends up to a given tie strength (A):
Click and drag the central circle to expand its radius to the desired size. All friends inside the circle are selected as post recipients.
Typical use case: only the friends up to a specific tie strength should receive the post, independent of the friend group. Example: a family visit to an amusement park. Although the information is suitable for all groups of friends, it is private information that might not be of interest for very distant friends.

Select friends of one or more friend groups up to a given tie strength (B):
Click and drag the inner rim of a wedge to create a wedge area and expand its size. As with (A), friends that are covered by the dragged wedge area are selected to receive the post. This process can be repeated with all available wedges. (A) and (B) can be combined.
Typical use case: only the friends in one or several friend groups, up to a specific tie strength, should receive the post. Example: pictures of a party at the university. The post is only suitable for a subset of the friends (most likely the friend group of fellow students). Additionally, the user might feel uncomfortable including distant fellow students in the recipients, and selects only friends up to a certain tie strength.

Exclude friends up to a given tie strength (C) and (D):
After a wedge area has been created in step (B), the user can click and drag the inner rim of the wedge area created in (B) to shrink it, excluding the inner friends of the wedge. As the shape of the wedge area becomes sickle-like by this process, this mode is later called "sickle mode"; the selected areas are denoted as "sickles".
Typical use case: only friends with intermediate tie strength of a friend group should receive the post. This is useful, for example, if you need feedback about something which is potentially embarrassing. Example: a rock band member asks for suggestions of some nice techno events. He might not want his best friends to know of his "special" hobby, but still needs some friends who know him well in order to receive good recommendations.

Select multiple areas inside a friend group (E):
Following (C)/(D), the processes (B) and (C)/(D) can be repeated in order to create a second sickle/wedge area. This process can be repeated up to four times per wedge.
Typical use case: A clique of friends inside a friend group should not receive the post. Example: you post a picture of fireworks that you bought in a foreign country. A sub-group of your friends would oppose such actions, and you want to exclude them.

Select/deselect single friends (F):
Independent of the selections in (A)–(E), a click on a profile picture selects or deselects the user as a recipient of the post.
Typical use case: a single friend has to be added to or removed from the audience due to special reasons. Example: a user posts a picture of her new partner on Facebook. To avoid conflicts, she wants to exclude her ex-partner from the audience.

Select all, select none
Two buttons to select all or no friends can be found in the lower left corner of the UI.
Typical use case: a post should be sent to all friends, or the user wants to reset the selection.

As stated in the introduction, the space in a radar interface is limited. For the use-case of selecting social network friends, we therefore had to implement techniques that allow us to display a larger number of friends inside the radar, which we will refer to as *space enhancement features* from this point:

A Incremental picture size Only a small subset of the Facebook friends are real friends that are of importance for recipient selection [66]. We therefore decrease the size of the friend images with decreasing tie strength, so that the closest friends gain the most importance.

B Zooming Using a double click, it is possible to zoom into and out of a certain area of the wedge to have a better overview, especially in crowded areas.

C Lighthouse design Based on the number of friends inside a wedge, OmniWedges selects some of the friend images (every second, third, fourth...) as lighthouse images that can be used as orientation points for the selection. All other friend images are shrunk to a small dot to avoid crowding.

D Overview bar When the user drags a wedge to select an area of friends, the overview bar is displayed to the right (see Figure 8.13): the line in the middle depicts the border of the current selection. Already-selected friends are listed below, including their name and friend images; not-yet-selected friends are displayed above. This functionality should help the user to spot exactly up to which tie strength he has currently selected, and whether more or fewer friends should be selected.

Technically, OmniWedges was implemented as a website with a frontend based on JavaScript, a Django backend and mySQL as a database engine.

8.2.2 User study

In order to validate the design of OmniWedges and to check whether the user interface improves the usability and the error rate of the friend selection process, we conducted a small-scale usability study at our university. The study was performed with 20 participants in a within-subject design, meaning half of the participants started with the Facebook interface and continued with OmniWedges, and vice versa for the other half of the participants. The participants were recruited by postings around the campus, and received a payment of 12 EUR for their participation.

At the beginning of the study, users had to first organize their friends into friend groups using the Facebook website. After the user claimed to be finished, we used automated scripts using the Selenium web toolkit[5] to traverse the user's Facebook website and to extract the created friend lists as well as the user names and links to their profile pictures. To maintain the data privacy rights of participants' social network friends, the procedure did *not* extract or store any profile pictures, but only hyperlinks to the friend's profile images that are later referred to by the OmniWedges UI.

As a last step before the main experiment, the user was presented with the friends sorted by ascending tie strength as calculated by Facebook, and had the task of adjusting the order if necessary. After a five-minute recovery break, the user was given 12 different tasks to solve for each interface. The first six tasks were of an *explicit* nature, meaning we explicitly stated *"select all university and family friends"* or *"select your 20 closest friends"*. For the remaining six tasks, we gave the subjects explicit posts with a sensitive nature like *"Please imagine you want to share pictures of a party that caused you to miss your family's Thanksgiving event"*. In order to achieve comparability of the results, each subject was given the same set of tasks and hypothetical posts instead of actual posts from the user's Facebook profile.

After each condition, the user was given a list of all friends for each task together with a "+" if the friend was selected as a recipient or a "-" if not. The user then had to go through the list and identify friends that were included by mistake (false positive) or mistakenly not included as a recipient (false negative).

The users had to fill out an AttrakDiff [150] usability questionnaire after each condition, resulting in four measures from -3 (worst) to +3 (best) describing the perceived attractiveness (ATT) of the interface, hedonic quality measures describing how stimulating the interface was (HQ-S) and how much they could identify with the UI (HQ-I), and a pragmatic score (PQ) similar to a system usability scale [150]. For the OmniWedges condition, the participants had to fill out an additional a questionnaire asking about the perceived *usefulness* of the *space enhancement features* on a seven-point scale (1=not useful at all, 7=very useful).

[5] https://www.seleniumhq.org/ (last accessed: 2020-03-09)

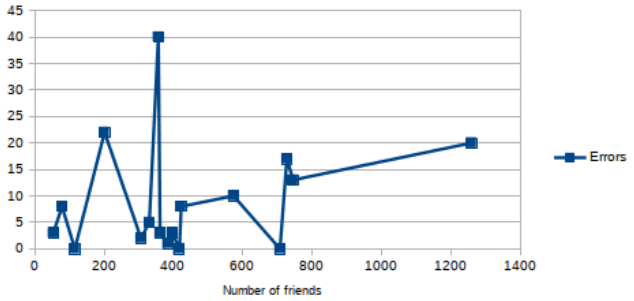


FIGURE 8.12: Comparison between number of friends and errors made using OmniWedges.

Before the experiment continued, the participants were given a five-minute break to rest and recover. The experiment closed with a semi-structured interview, where we tried to find out which interface they preferred, which selection strategy they used with OmniWedges, whether their posting behavior might change and which additional functionalities or combinations of interfaces between Facebook and OmniWedges were perceived as useful. The procedure was reviewed and approved by the ethical review board of our institution.

8.2.3 Results

The age of the participants ranged from 21 to 38 years (mean 24.38). 66% of the 20 participants were female, 44% male. The number of Facebook friends of the participants was between 53 and 1260 (mean 437), representing a good portion of average user profiles[6]. As the number of friends was very variable, we had to normalize the number of false positives and negatives by dividing them by the number of friend list entries before starting the analysis. We performed a 2 (condition) x 2 (explicit or implicit task) x 2 (false positive or false negative) ANOVA to compare the errors made throughout the study. If only the interface is taken as an effect, the results show that using OmniWedges significantly decreased the amount of errors ($F = 5.57$, $p = 0.031$, $M_{Wedges} = 0.0076$, $M_{Facebook} = 0.020$). Whether the task was an explicit or implicit task did not have any significant effect on the error rate between the two interfaces ($F = 0.677$, $p = 0.423$ using interface and task type as effects). The type of error that was made (false positive or false negative) did not depend on the interface ($F = 0.001, p = 0.98$, $M_{FP-Wedges} = 0.0099$, $M_{FN-Wedges} = 0.0052$, $M_{FP-FB} = 0.0221$, $M_{FN-FB} = 0.0176$ using interface and error as effects). Taking a look at the comparison between the number of friends and the amount of errors made in OmniWedges in total (Figure 8.12) does not show any disproportionate increase in errors with an increasing number of friends. A correlation analysis between number of friends and errors made (normalized) was not significant ($r = -0.370, p = 0.144$), indicating that the number of friends, and hence the degree of population of the UI, does not have any significant effect on the errors made.

The *space enhancement features* were perceived as very useful by the users, as shown in Table 8.4. The Overview bar was voted as the most useful feature with a score of 6.38 on average on a scale from 1 (worst) to 7 (best). The lighthouse design and zooming achieved similar scores of 5.81 and 5.9, identifying them as the second

[6] http://blog.stephenwolfram.com/2013/04/data-science-of-the-facebook-world/ (last accessed: 2020-03-09)

feature	mean	stdev
Overview bar	6.38	1.284
Incremental picture size	5.43	1.248
Lighthouse design	5.81	0.981
Zooming	5.9	1.179

TABLE 8.4: Perceived usefulness of the *space enhancement features* of OmniWedges.

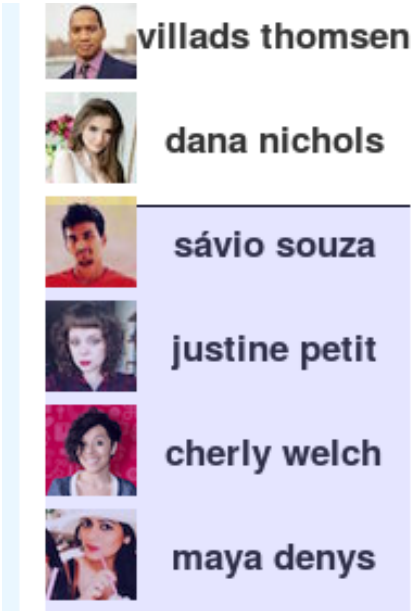


FIGURE 8.13: Overview bar during selection process.

most useful features. Finally, the incremental picture size recevied an average score of 5.43, which was the lowest of all scores, although it was still perceived as very useful.

We calculated the attractiveness (ATT), pragmatic score (PQ) and the two hedonic scores (HQ-S and HQ-I) by calculating the average over the responses for the corresponding questions in the AttrakDiff questionnaire [150] and performed four paired T-tests to spot significant differences between the two interfaces. OmniWedges outperforms its Facebook counterpart in terms of attractiveness ($T = 6.115, p < 0.001$, $M_{Wedges} = 5.37, M_{Facebook} = 3.35$) as well as the hedonic quality ($T_{HQ-I} = 4.93, p_{HQ-I} < 0.001$, $M_{HQ-I-Wedges} = 5.09, M_{HQ-I-Facebook} = 3.66$; $T_{HQ-S} = 7.83, p_{HQ-S} < 0.001$, $M_{HQ-S-Wedges} = 5.26, M_{HQ-S-Facebook} = 3.17$) with high significance, assuring a better user experience. There is also a tendency for a higher pragmatic quality using OmniWedges ($T = 1.83, p = 0.082$, $M_{Wedges} = 4.75, M_{Facebook} = 4.06$), although we could not prove a significance here.

In the semi-structured interview, 85.7% stated they would use OmniWedges if it was integrated into Facebook. 61.1% would use it only if they had a post that would be of interest only for some friends ("narrowcasting") instead of the "custom privacy settings" view on Facebook, or if it contained sensitive information (just like the example tasks in our study). 85.7% stated they would change their posting behavior

when using OmniWedges. Of those who would, 88.9% would do more narrowcasting, and the remaining 11.1% would post more sensitive posts. When asked explicitly, 42.9% of the participants stated they would post more on Facebook when using OmniWedges, 38% would not do so, and 19% remained unsure. When asked which interface they preferred and which functionality they would add, 50% stated they preferred OmniWedges with an additional search field for finding specific friends, 33% preferred a combination of OmniWedges and the Facebook interface, and 8.3% wished to keep only OmniWedges or the Facebook interface, respectively. When asked about what selection strategy they used within OmniWedges, all participants used one of two different strategies, or a combination thereof: In the first strategy, users only searched for single persons they wanted to select (3 participants). In the second strategy, the participants thought more of areas inside the wedges they wanted to include, rather than single persons (10 participants). Finally, 7 remaining participants used a combination of both strategies, where they first made a raw selection using the wedges, which was then refined by selecting or deselecting single persons in a second step.

8.2.4 Discussion

Scalability

As stated in the introduction, the space in a radar interface is very limited. Without modifications, this approach is not suitable for a social network, where users can have up to thousands of friends that are potential receivers of newly created posts. We therefore introduced several UI mechanisms (see Section 8.2.1) that allow the display of all of a user's friends inside our radar interface. The results of the study show that, using these improvements, the concept of a radar metaphor can also be used to display a large number of friends while still reducing the amount of errors made during the selection process. The correlation analysis showed that there is no significant effect on the number of errors made using OmniWedges depending on the amount of friends displayed, indicating that the techniques achieved their goal. However, although the techniques seem to work in this application context, we are interested in whether these ideas could be transferred to other domains or use cases, which we are eager to research in future work.

User acceptance & UI improvements

According to the interview results, a large majority of users would replace the current audience selection method on Facebook with a version of OmniWedges. Nevertheless, there are still some improvements that were suggested by the users and that are perceived as must-haves if OmniWedges is to be used on a daily basis. About half of the participants would use the UI only if a search functionality was introduced, which would help them find single, critical friends that they want to include in or exclude from their selection. One third of the subjects preferred a combination of OmniWedges and the Facebook interface, so they could select larger portions of friends using OmniWedges and easily find and select single friends using the Facebook interface. Although they declared this combination to be their favorite, we cannot clearly state why they voted for the additional integration of the Facebook UI: whether they only need it as a kind of search functionality for OmniWedges or whether they are also interested in other functionalities that cannot be included in OmniWedges. Whether we can also satisfy these users' desires with an improved

OmniWedges interface, including a search functionality, should be researched in future work.

Change in mental model

Although the procedure and tasks were the same for both interfaces, the interview answers indicate that only the OmniWedges interface changes users' mental model, compared to the standard Facebook interface: Most users tend to do more narrowcasting, i.e. disclosing posts only to people who might be interested in them. A smaller number of the subjects also stated they would publish more sensitive posts with our interface. The answers lead to the assumption that the different kind of visualization that we use leads to a different awareness of the post audience: Rather than always displaying only a small portion of all friends at once in a scrollable list, the radar interface displays *all* of a user's friends at once, allowing them to have an overview of the large number of friends that would see the post when the sharing option is set to "all friends". Therefore, users begin to think about whether this large audience is really the desired audience for their post, resulting in a more rigorous limitation of the post audience with OmniWedges, and therefore a smaller amount of false positives and false negatives, as the study results show. On the other hand, users seem to have a higher trust in their audience selection when they can see all of their friends at a glance, leading them to post sensitive posts more frequently with OmniWedges. However, the aforementioned motivations for different behavior are only guesswork. Proving their correctness remains as future work.

Observed selection strategies

The change in the mental model is also reflected in the different usage strategy when using OmniWedges compared to Facebook. Where Facebook only supports the *single selection strategy*, where the audience can be refined by adding a single person or a group of friends, OmniWedges also allows users to select areas containing multiple friends with a similar tie strength. In fact, most of the users (70%) used the *area selection strategy* either as their sole selection strategy, or in combination with the *single selection strategy* for refinement. But still, a small amount of users (15%) did not take advantage of the *area selection strategy* at all. In future research, we would like to elaborate on the efficiency of the different strategies, e.g. which one leads to the most false positives/negatives, which is the least time-consuming strategy, and whether there is an optimal strategy that optimizes both factors at the same time.

Application to historical posts

OmniWedges was designed to select the audience for a new social network post. Nevertheless, the same design can also be used to visualize and review the audience for historical user posts that have been published in the past, even if a different user interface was used for the selection: selected users are initially selected as single friend selections (action F in Figure 8.11). If the UI detects two or more selected friends next to each other inside the same wedge, a selected area is created around them (actions B to E). By that means, OmniWedges is able to construct a selection that corresponds to the historical sharing setting using the wedge-based UI. We would like to examine the usefulness and correctness of such a functionality in future research.

8.2.5 Conclusion

Most social network posts are still shared with "all friends" although this mostly does not correspond to users' actual sharing desires and often leads to unwanted disclosures. Radar interfaces have been proven to increase privacy awareness and thereby to motivate users in adapting their privacy settings, leading to a reduced amount of unwanted disclosure and an increased trust in the privacy settings. We adapted the radar approach to offer a radar-based audience selection tool for social networks, called *OmniWedges*. We integrated several techniques to be able to show huge numbers of social network friends in our UI, while still maintaining usability. The study results show that OmniWedges significantly decreases the error rate and offers a higher usability. In a semi-structured interview after the study, the results indicate that the different UI design also leads to a different mental model on the part of the users, which leads to an increased awareness of the amount of recipients selected, and in turn leads to a more frequent use of recipient limitation techniques like narrowcasting. The UI also seems to increase trust in the privacy settings, which leads users to post more sensitive posts. Whether the UI can be succesful when it is integrated into daily use of a social network remains to be proven in an in-the-wild study in the future. Whereas OmniWedges was meant to define the audience for a data item like a social network post, the next section will discuss an approach that is targeted towards defining the *granularity* of information that should be shared, considering the example of the intelligent shopping domain.

8.3 Assisting users in detecting flaws in their privacy settings using a privacy overview

Especially after the introduction of the General Data Protection Law (GDPR) in the European union, data privacy has received increased attention also in (online) retail business[7]. Whereas data was collected in an aggregated way or at least anonymously, intelligent retail stores like Amazon Go[8] have to collect a lot of personal data to be able to offer their assistence systems, like the invisible checkout system that already knows the items inside the customer's basket before she arrives at the checkout. Upon entering the shop, the shopper uses the NFC functionality of her smartphone to identify himself at the entrance gate. She can then browse the store, grab products, put them back again, and leave the store with the selected products, without a checkout or scanning process. Amazon achieves this using "sensor fusion and deep learning"[9] without naming further details. The technology behind the service is most likely based on camera systems and a combination of different sensors, and tracks the customer throughout the complete shopping process, from entering the store and identifying herself through the smartphone, through viewed products and items placed in the shopping basket, to the invisible checkout, that allows the customer to leave the store without any payment procedure, by withdrawing the price of the items registered in the shopping basket from the user's credit card. On one hand, the instrumentalization of this intelligent retail store allows the customer to save a lot of time. On the other hand, customers also mention privacy concerns: in order to make the service work, Amazon has to record and store a large amount of private data throughout the shopping process, and it is not even anonymized. Which sensors are used, which data they generate, and in which form they are stored, anonymized or deanonymized, is not clear to the user.

Apart from operative intelligent retail stores like Amazon Go, there exist several research laboratories like the Innovative Retail Laboratory (IRL) [308], which investigate the capabilities of new technologies in the context of brick-and-mortar retail stores. The Innovative Retail Laboratory (IRL) is an application-oriented research laboratory of the German Research Center for Artificial Intelligence (DFKI) run in collaboration with the German retailer GLOBUS SB-Warenhaus Holding in St. Wendel. In this living lab, research is conducted in a wide range of different domains, mostly related to intelligent shopping assistance. The demonstrators range from an instrumented shopping cart employing indoor navigation to several intelligent shopping consultants, ambient information services and an automated checkout system.

Until fairly recently, such features have only been concepts and never implemented in a real store. Maybe this is one reason why the research field of retail data privacy has been mostly neglected in the past: related work has done research in many other fields where sensitive data plays a role, like social networks [180], location sharing [169] or mobile app permission setting [206]. However, the application of those approaches to data from the intelligent retail shopping domain has not been a subject of research so far. The current standard in the aforementioned domains is list-based interfaces, where the permission settings are listed in sequence together with an option to adapt each setting. Although such an interface give the

[7] https://www.bdo.com/insights/industries/retail-consumer-products/gdpr-what-retailers-need-to-know-about-the-new-e (last accessed: 2020-03-09)

[8] https://www.amazon.com/b?ie=UTF8&node=16008589011 (last accessed: 2020-03-09)

[9] http://www.self.com/story/amazon-go-grocery-store-of-the-future (last accessed: 2020-03-09)

user the freedom to adapt any of the given settings, those interfaces neither offer a good overview, nor are they perceived as attractive and motivating [67], which leads to the fact that the task of choosing privacy settings is perceived only as burdensome and "not worth it" by most users [128, 220]. Therefore, users typically stick to the standard privacy settings. Christin et al. provided a more sophisticated interface based on the radar metaphor, called *privacy radar* [67], that visualized the privacy settings for participatory sensing applications. The authors conducted a study on the effects of their privacy radar, and found out that their design led to a significantly better overview on the privacy status, and thereby motivated users in adapting their privacy settings [67].

In this section we want to examine how a conventional list-based interface as well as such a radar interface could look like in the context of retail data. Our primary goal is to create an interface that is enjoyable and more fun to use than a conventional list-based interface, in order to motivate users to use the interface more frequently. Although list-based interfaces are already very efficient to use, we tried to develop an alternative UI that is at least as efficient and fast to use as a conventional list-based interface. In order to develop the interface, we reuse the results from the background research in Chapter 6, Section 6.2, where we already investigated *which* data is and could be used inside an intelligent retail store, and how the data types can be *clustered* and sorted into groups, in order to structure the user interface accordingly.

In detail, we try to solve the following research questions:

1. How do the interaction times of the radar interface compare to those of a list-based interface?
2. Can we make it easier to detect unusual privacy settings using an interface based on a radar metaphor?
3. How convenient is the radar interface compared to the list interface?
4. Does a radar interface for retail data provide a better user experience than a standard UI?

The next sections will first describe similar approaches in other domains, and afterwards present the implementation of the list and the radar interface based on prior background research discussed in Chapter 6. Afterwards, we will describe the evaluation study, and a discussion of the results. The work presented here is based on previously published research [267].

8.3.1 Retail privacy user interfaces

We propose *two* interfaces to control the disclosement of shopping data, the first based on a conventional list-based interface, and the second inspired by a radar metaphor [67], which was also used as an inspiration for the audience selection UI discussed in Section 8.2. Each interface has two major tasks: On the one hand it should give a clear overview on the settings, and allow the user to set his privacy settings with a minimum of effort. On the other hand, it should be easily possible to compare one's own settings with the settings of an average user, and to quickly detect unusual decisions. In Chapter 6, we already described a background study to find out the data types recorded inside an intelligent retail store, as well as the services and the personal data needed by the service in order to operate. For details on the identified data types, data groups and orderings, please refer to Chapter 6,

Section 6.2. To quickly summarize the results, we worked out that there are four major different parties that might be interested in retail data and that we wanted to include in the UIs, later called *stakeholders*: friends and family of the shopper, the intelligent retail store that records the data, and third parties that might be interested in creating a shopping profile to offer personalized ads to the shopper. There are several types of data that could be recorded in an intelligent retail store (see Table 6.1), like personal information, along with location data, in-store movement profiles, or items viewed, that can be grouped into four distinct *data groups*. Except for one of the data groups, there is a sensitivity order for the data in each data group that is shared among most users. The next two subsections will describe the two created privacy UIs.

List-based interface

Privacy interfaces have been widely used in other domains like social networks, location sharing and mobile app permissions. In most cases, the privacy settings can be manipulated using a list-based interface, i.e. the different data types (photos, posts, comments, locations, app permissions, etc.) are listed along with a button or slider to select whether and from whom the data should be kept private. Figure 8.14 shows the *list-based interface* of *URetail*. The list interface contains one webpage for each of the four *stakeholders*. The navigation bar on the left side (1) of each page allows the user to switch to the pages of the other stakeholders.

Except for the background color (and the headline) that distinguishes the four different pages, all of them are designed in the same style. The different data types are listed one after the other, grouped according to the groups of data types acquired in Chapter 6. To the right of each data type entry, a draggable slider can be found, which allows the user to change the disclosure policy to disclose/not disclose (2).

In addition to the list of data, we want the user to have an overview on which data types he has shared compared to an average user. For this reason, a summary of his disclosure choices along with an average disclosure setting is displayed at the bottom of each group (3), later called the *status message*. The numbers on the left side of the fraction represent the user's amount of disclosed data in this group, and on the right, the average amount of data disclosures.

Radar interface

Although the list-based interface is widely used, the immense number of switches is often seen as very confusing by the average user; furthermore, it is hard to see which settings are unusual, or possibly misconfigured [277]. Christin et al. succesfully used a radar metaphor to visualize privacy threats in participatory sensing applications. Inspired by this interface, we created a privacy interface based on a radar metaphor as depicted in Figure 8.15: the different data types are grouped together in six different wedges. The data items of the six data groups can be ordered by ascending sensitivity of the data. Each of the wedges contains several layers, corresponding to the data types inside the corresponding group in ascending order from the inside to the outer rim. Similar to OmniWedges, the user has to click and drag from the center of the wedge to the outer rim, to select the data types to be shared up to this point (1). To share no data from the given group, the user has to choose the innermost layer. Each wedge has a corresponding list view next to it, which displays the data items in the same order as the wedge's layer (2), together with a checkmark if the data type is shared or a red cross if it is not. It is also possible to adapt the order

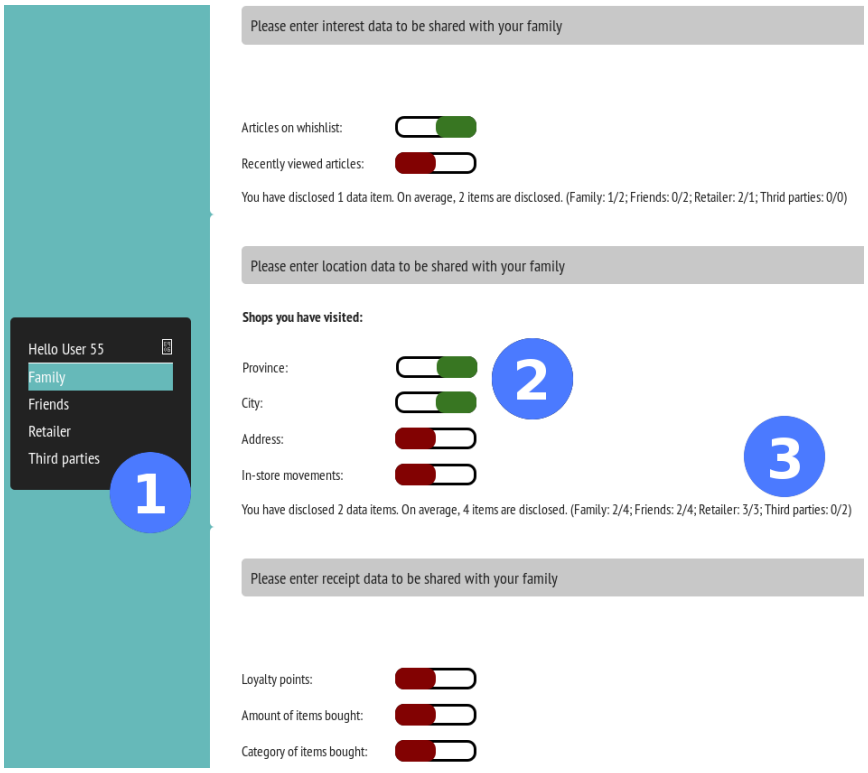


FIGURE 8.14: List-based interface of URetail: All data types are listed one after the other along with a slider to change the disclosure setting.

of layers using drag and drop in the list, or to enable or disable the sharing of single data types by clicking on the respective item. The radar interface supports the display of several stakeholders, realized by several radars that can be selected using the navigation bar on the left, similar to the list interface.

In addition to displaying the three radars seperately, the radar interface also offers a 3D overview ("privacy pyramid") at the bottom of the UI (3) that shows all four radars at once. This privacy pyramid consists of four layers stacked on top of each other, each of the representing one of the four stakeholders. Within each layer, the data groups are represented by the four edges. The larger the distance from the edge to the center, the more data types are disclosed. Usually, the least information is shared with third parties, more with retail companies, and the most with family and friends, resulting in the pyramid-like shape of the privacy pyramid. To get an impression of how an average profile looks, the user can display the privacy pyramid of an average user as a transparent overlay (see Figure 8.16).

8.3.2 Evaluation

Although both UIs were designed in a user-centric way involving user feedback cycles, we checked the usability in a final evaluation study. The interfaces had two main goals: first, to allow management of the disclosure settings, and second, to give a clear overview on the settings, and to be able to compare them to those of an average user. The management part was realized with a list or a radar interface, whereas the second goal was achieved with a simple status message in the list interface, and a three-dimensional privacy pyramid in the radar interface. The goal of the study was therefore to validate the results of the background research and

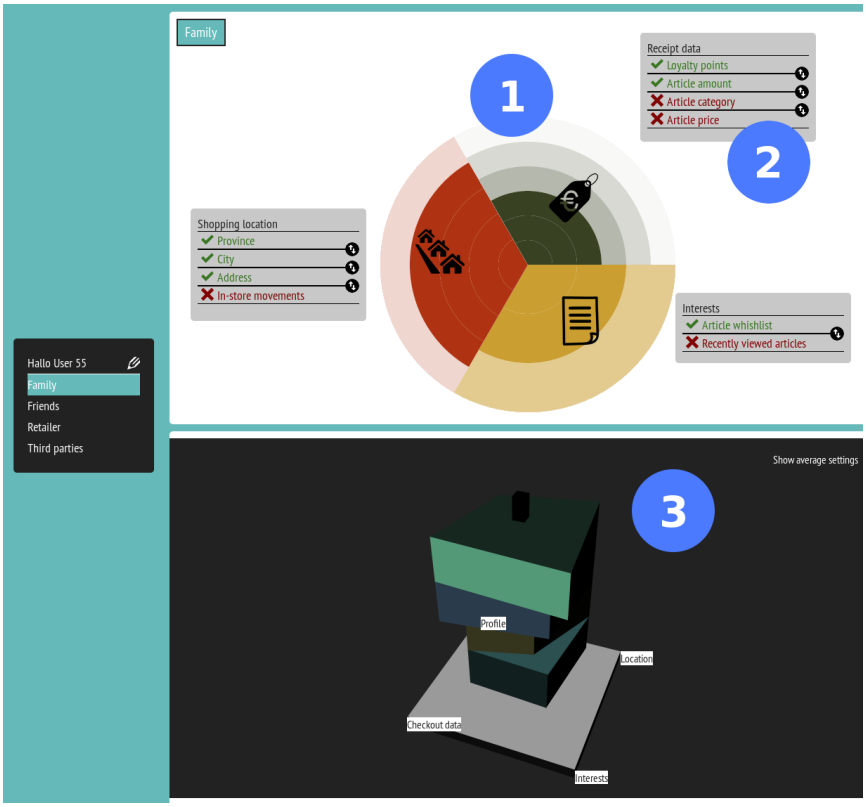


FIGURE 8.15: Radar interface of URetail: data types are arranged in groups in a circular form, sorted by the sensitivity of the data types.

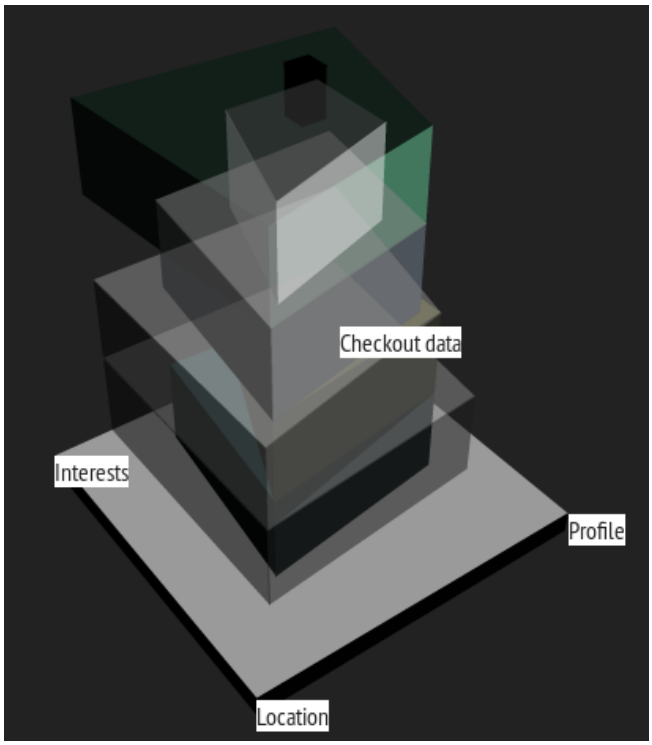


FIGURE 8.16: Privacy pyramid: comparison to the average user.

Label	Question
	It was easy for me to spot:
whether	- whether there are unusual settings
how_much	- how many are unusual
which	- which settings are unusual
average	- what an average value is
diff	- how much the difference from an average value is
	I found the visualization to be:
clear	- clear
appealing	- visually appealing
fun	- fun to use

TABLE 8.5: Practicality questionnaire used in the evaluation.

the radar interface concept and to compare for both interfaces the interaction times, practicality and usability in terms of hedonic and pragmatic quality.

The study was conducted as a lab study with 21 participants from the university context (students of different subjects). Their ages ranged from 17 to 52 with an average of 31.19 years. The study was designed within-subject, which means that each subject used both interfaces, one after the other. To prevent biasing of the results, the order in which the participants used the two interfaces was decided using a Latin square, i.e. half of them started with the list-based, the other half with the radar-based interface. The procedure was the same for all participants, and started with a questionnaire capturing demographic data. Afterwards, they were given an introduction in the first interface condition (either the list-based or the radar-based one), explaining all functionalities and displayed UI elements, and had some time to try out the UI themselves. After they claimed to be sufficiently familiar with the interface, they were given the task to set all the privacy settings according to their needs. The time measurement started directly after the task instruction, and ended with the last interaction. For the radar interface, we also recorded when single items were (de-)selected using the lists ((2) in Figure 8.15), and how often the items were rearranged. After succesful completion of this first task, we started over with the second one: we loaded a privacy profile that had some unusual settings, which had to be spotted using either the privacy pyramid or the status messages. The subjects had to find out and state *where* the unusual settings were, and to *adapt* them so they matched the settings of an average user. We again recorded the interaction time from the first to the last click, and whether all unusual settings were correctly found and changed. Each condition ended with two questionnaires: first the AttrakDiff questionnaire, which captures the hedonic and pragmatic quality of the interface, and second, a custom questionnaire to capture which interface allows an easier completion of the comparison task, using either the privacy pyramid or the status messages. The statements on the custom questionnaire (see Table 8.5) were recorded on a five-point scale from fully agree (=1) to fully disagree (=5). The procedure was immediately repeated for the second interface. The experiment ended with an informal discussion of the perceived problems and advantages of both interfaces, possible improvements and other thoughts about the experiment.

To conclude, we recorded the following measures:

1. Interaction time with the radar or the list interface for the first task (adjust disclosure settings)
2. Frequency of list selections in the radar interface
3. Rearranging frequency in the radar interface
4. Interaction time for the second task (find deviations from average settings and change them) with the privacy pyramid or status messages
5. Number of deviations that were not spotted
6. Practicality of the privacy pyramid and status messages using our custom questionnaire
7. User experience and usability using the AttrakDiff questionnaire
8. Qualitative feedback from the interview at the end of the experiment

Results

Only 11 out of 20 participants selected single items in the radar interface. Those who did clicked on average on two items throughout the whole experiment, over the four pages of the stakeholders. Rearranging was used by three participants; on average two item switches were done. Two out of three participants dragged the loyalty points to the outer rim of the radar, so it became the item with the least sensitivity. The low frequency of single selections and rearrangements supports the results of the study in Chapter 6, except for the loyalty points item, which seems to have a different sensitivity level for a small portion of the participants.

All participants spotted all deviations from the average settings profile with both interfaces in the second task. In contrast, the results of the custom practicality questionnaire were quite diverse, as listed in Table 8.6. For the analysis of the questionnaire data, we first performed Shapiro-Wilk and Kolmogorov-Smirnov tests to ensure normality, and used a paired T-test later, as these tests were positive. Remember that the scale ranges from 1 (=best) to 5 (=worst). Users found it significantly easier to spot *whether* there are unusual deviations from an average settings profile ($M_{radar} = 1.62, M_{lists} = 2.14, p = 0.012$). In contrast, the list interface is better for seeing how an average profile looks, and what the differences are ($M_{radar} = 2.19, M_{lists} = 1.67, p = 0.012$). Although the radar interface on average performs better in spotting how many differences exist ($M_{radar} = 2.48, M_{lists} = 2.57, p = 0.733$) and which things are different ($M_{radar} = 2.29, M_{lists} = 2.62, p = 0.217$), we were not able to prove a significance within our test set. The visualization of the radar interface is on average seen as more clear, but not with a significant difference ($M_{radar} = 2.10, M_{lists} = 2.29, p = 0.296$). Furthermore, it is perceived as significantly more visually appealing ($M_{radar} = 2.15, M_{lists} = 3.10, p = 0.001$) and fun to use ($M_{radar} = 2.14, M_{lists} = 3.19, p < 0.001$), which is an important factor in a boring, burdensome task like choosing privacy settings. These results are also reflected in the AttrakDiff scores (Table 8.7), which clearly favor the radar over the list-based interface. The scores for each measure range from -3 (worst) to +3 (best). An average user interface would have a neutral pragmatic and hedonic score (about zero) and would therefore be located in the center of the central square (neutral) of the portfolio presentation (Figure 8.17).

FIGURE 8.17: Portfolio presentation of the AttrakDiff results, including the average hedonic (HQ) and pragmatic quality (PQ) as well as the confidence rectangle. Pragmatic quality increases to the right, hedonic quality to the top of the graph.

Measure	List	Radar	T	p
Whether	2.14	1.62	- 2.75	.012
How_much	2.57	2.48	- .346	.733
Which	2.62	2.29	- 1.276	.217
Average	1.81	2.62	3.068	.006
Diff	1.67	2.19	2.95	.008
Clear	2.29	2.10	-1.073	.296
Appealing	3.10	2.14	-3.76	.001
Fun	3.19	2.14	-5.966	<0.001

TABLE 8.6: Results of the custom practicality questionnaire.

Scores > 1 or < -1 are perceived as *above average* or *below average* respectively [150]. As the tests on normality failed this time, we used a Wilcoxon signed ranks test, which is the non-parametric equivalent of a paired t-test. All four measures attractiveness (ATT), hedonic quality regarding stimulation (HQ-S) and self-identification with the interface (HQ-I), and pragmatic quality (PQ) are better for the radar interface, with high significance. All mean scores for the radar interface are above one and, except for PQ, also above two, attesting to an excellent user experience (HQ) and a pragmatic quality (PQ) clearly above average. The list-based interface, on the other hand, has a pragmatic quality which is somewhat above average, but a user experience notably less than average. The stimulation (HQ-S) in particular is clearly below average, classifying it as a significantly less interesting interface than the radar.

Although the radar offers a significantly better user experience and is also perceived as more practical, we could not prove that there was a significant difference for the interaction times between the two interfaces. We conducted Shapiro-Wilk and Kolmogorov-Smirnov tests to succesfully prove the normal distribution of the interaction times (1. and 2.). Therefore, we were able to use a paired T-test on these measures, whose results can be found in Table 8.8.

The radar interface, as well as the privacy pyramid, had on average a similar interaction time as the list interface with its status messages. The qualitative feedback from the closing interview, as well as the implications of the results above stated, will be discussed in the next section.

	List	Radar	Z	p
PQ	0.69	1.99	-3.785	<0.001
HQ-I	-0.16	2.18	-4.015	<0.001
HQ-S	-1.33	2.38	-4.017	<0.001
ATT	-0.31	2.21	-4.026	<0.001

TABLE 8.7: Results of the AttrakDiff questionnaire.

	List	**Radar**	**T**	**p**
Adjustment task	156.76	151.10	-1.005	0.327
Find task	75.57	70.81	0.891	0.383

TABLE 8.8: Statistical values for the interaction times in milliseconds.

8.3.3 Discussion

Feedback and improvement suggestions from the interviews

Besides some minor design suggestions like coloring of the pyramid, or the order of the loyalty points in the radar interface, the interview at the end of the experiment brought some interesting thoughts to light: some participants disclosed more data to retailers and third parties than to friends and family. When asked for the reason for that decision, they stated that it is easier for them to disclose the data to an impersonal entity like a retailer or a third party company, than to a person they know personally. One person stated that this is caused by the fact that he has a specific image that he wants to keep for the people he knows. For example, if a shopper has a masculine image in his circle of friends, he might not want them to see that he just bought some knitting accessories. Disclosing the wrong information might disturb that social role, and therefore he is more cautious with disclosing that data to well-known persons. He also suggested to include a functionality in the radar interface that allows users to disclose data types according to the role of that product. The receipt data, for example, could be filtered before disclosure to friends and family, to maintain a personal image towards that group of persons. In the example above, the user would give the UI the information that he wants to be seen as masculine by his friends. The interface then matches from the role to products that most likely violate that role, e.g. knitting products or ballet accessoires, and hides them. Furthermore, most people stated that the radar interface is at a first glance harder to understand and use, but becomes more convenient than the list interface after the first training session and throughout the evaluation tasks. They stated that they were more used to the list interface, and could therefore use it directly without any problems. Nevertheless if they used the radar interface more frequently, especially in everyday use, they would most likely prefer that interface, and also be able to work with it faster than the list.

Validation of earlier results

The studies on data types in the intelligent retail domain, groupings and severity orders in Chapter 6 were only meant as an explorative study to get a first impression on how the data could be clustered, and which clusters could be ordered according to the sensitivity of the data. Therefore we recorded in the evaluation study, for all participants, whether they checked single data items in the list or whether they

changed the order of items in the clusters. As the evaluation study results show, single items rarely got selected or had their order switched. Mostly, the loyalty points were moved from the position of the least sensitive item (innermost layer) to the most sensitive item (outermost layer). This effect has to be further observed; perhaps the loyalty points are perceived very differently for a larger group of users, and might be realized as a slider inside the list in a future interface. Aside from this outlier, the clusters and order of the data types inside them could be approved by the evaluation study, supporting the reasonableness of our radar interface concept.

User experience differences

Both interfaces had an acceptable pragmatic quality, while the radar interface was still significantly better than the list. Although both interfaces are therefore handy to use, the list is anything but an appealing and engaging interface. The stimulation (HQ-S) in particular is very low, which also means that the fun factor is very low. As earlier research has already shown, the fact that setting privacy settings in social networks is a very tedious task leads users to almost never touch their privacy settings page [128, 220]. The radar, on the other hand, has an outstanding user experience, especially regarding stimulation. Combined with the increased practicality (PQ), it may lead the users to open and adapt their privacy settings more often, or at least reduce the resistance to doing so.

Similar interaction times for the privacy setup

Although the main focus of our work was to design an interface which enhances the user experience and enjoyability while maintaining one's settings, we were also interested in the interaction times of the two interfaces. The radar interface in fact needs fewer clicks to complete all the settings (only one per group vs. many per group for the list), and is also perceived as more practical, easier and faster to use. Interestingly, the interaction times are not significantly different. There are two possible reasons for that mismatch: First, the list interface splits the disclosure decisions into smaller, easier tasks, as there is one slider for each data type that has to be set. The time needed for the disclosure decision is therefore the same, whether they go through the list and check the items to disclose, or whether they go though the data types in the radar interface, and decide where to make the cutoff. Furthermore, people are more used to conventional list-based interfaces, as they are widely used. This also leads to the second possible reason, namely that radar-based interfaces are quite new and have to be learned first to be used efficiently. Although we gave the users some time to get familiar with both interfaces, a definite answer can only be given after several consecutive training sessions, which would be a good starting point for future work.

Privacy pyramid vs. status messages

The results for the second task, namely spotting unusual settings, brought quite diverse results to light: a graphical 3D interface like the privacy pyramid allows users to spot very quickly *whether* there are unusual settings, but it is easier to identify the differences and the normal values with a text-based interface like the status messages. The quantity of differences and which things are different seem to be seen more easily with the pyramid, although we could not prove a significance within

our small sample set. This implies that neither a 3D interface, nor a textual interface is best for this kind of task. Only a combination of both interfaces can perform optimally. Whether this assumption holds should be verified in future work.

8.3.4 Conclusion

The rise of intelligent retail stores like Amazon Go makes it necessary to offer shoppers a control interface where they can determine which of their personal data should be recorded and disclosed to the retailer, third parties, or other stakeholders. Current list-based privacy interfaces, best known from social networks or mobile app permission settings, are on one hand very efficient, but on the other hand neither visually appealing, nor fun to use, leading to the fact that they are almost never used. We presented two user interfaces, a traditional list-based interface and a more modern radar-based interface that should enhance the user experience compared to the former approach. The succeeding evaluation study showed that the radar interface provides a user experience which is clearly superior to a conventional interface, and that is perceived as more practical, although the interaction times are similar on the first try. Nevertheless we found some interesting points on how the combination of both interfaces could lead to an even stronger UI, and spotted interesting directions for future work in the experiment interviews and the discussion of the results.

After having described the user interface design for defining the audience and also for defining the detail level of the data to be shared, the last chapter will now discuss how in-situ feedback can be collected and used for refining already existing privacy settings.

8.4 Leveraging in-situ user feedback for privacy settings

As already stated in earlier chapters, the perceived audience of a social network post comprises only a small amount of the actual audience, on average 27% of its true size. Research has already begun to tackle this problem by giving users an overview of their audience and the possibility to withdraw their post before the final publication [333]. Another problem that arose recently with the redesign of the Facebook newsfeed is a mechanism called *algorithmic curation*, which organizes and selects the posts that should appear on the user's news wall. The majority of users (62.5%) are not aware that the posts on their news wall have been filtered and selected previously, and typically feel anger when being informed about their existence [270, 100]. A study has shown that if users have the possiblity to control which posts are filtered and how, they are significantly more engaged in using the social network and have a higher feeling of control on the website [100].

Selecting the correct post audience is a task that is often neglected by users due to its high complexity and the additional effort that is needed for every post [128, 218]. Therefore, most posts are shared with "all friends" instead of a targeted audience [218]. One approach that can assist the user in selecting her privacy settings, namely by proposing privacy settings based on the user's individual measures, has already been discussed in earlier chapters. Nonetheless, those systems require the user to be intrinsically motivated in using the system, and still need the user to invest some time in choosing the settings, although this time could already be reduced using machine learning support. Related work has already explored an approach called "Twitch Crowdsourcing" that motivates users in doing crowdsourcing tasks by fostering the common habit of turning to one's mobile phone in spare moments, by providing short micro-tasks with a duration of one or two seconds whenever the screen is unlocked [328]. Within this section, we discuss whether a similar feedback method, namely in-situ feedback on social network update notifications, can work as a possible solution for both of the aforementioned problems.

For the study, we implemented a mockup smartphone app, which adds a feedback functionality to social network posts, as shown in Figure 8.18. Below the notification, we added two buttons allowing the user to give either positive or negative feedback. Negative feedback means that the user *disliked that the user was able to see* and like the post, meaning he experienced some kind of privacy invasion; whereas positive feedback means that the user *liked* that the friend could see and like the post. If the user does not want to give any rating, she can also decide to swipe away or to delete the notification as usual. As stated earlier, our goal is to catch in-situ feedback on the go, wherever the user is; therefore we designed the interface to be as simple and quick to use as possible [328], offering only a binary feedback choice to the user.

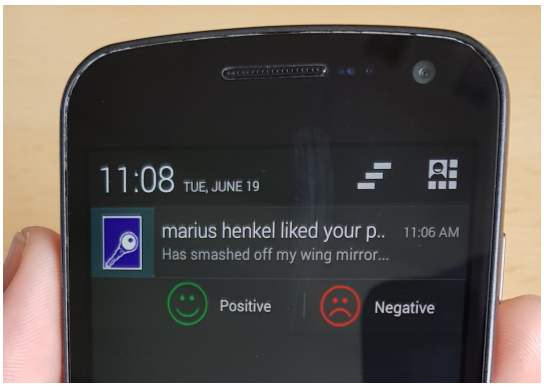


FIGURE 8.18: Smartphone showing a Facebook update notification using our application, providing buttons for in-situ feedback.

Within the described approach, we faced several interesting questions that we try to answer within this section:

1. How do users currently react to unwanted comments or likes?
2. Which possible solutions can users imagine for this problem?
3. When using in-situ feedback for the given problem, how could the possible feedback options look? Should these look like a "positive" or "negative" decision, as shown in the UI example? Do users prefer other answers, or is it better to use 3, 4 or more different options?
4. Which effect on the post audience or the news feed is expected by users, if one of the two buttons is pressed? Do they expect a direct impact on the post audience or the news feed?
5. What other uses of the feedback can users imagine?

In order to solve these questions, we conducted a small-scale qualitative study with four focus groups. Throughout this section, we will first take a look at how in-situ feedback is used in other scenarios, then give a detailed description of the conducted focus groups and the findings that we gathered throughout these sessions. We will discuss the findings and assemble them into a new approach later on that is capable of managing privacy settings and that allows content elicitation using in-situ feedback, as described in the "Outlook" section. The work presented here is based on already published research [263].

8.4.1 Study methodology

As stated in the introduction, the goal of our work is to find out whether in-situ feedback is useful for social network notifications, and what effects users expect after giving feedback. For this purpose, we created a mockup of a part of the Facebook mobile phone application that allows users to give in-situ feedback for each update notification. The app is not fully functional, as social networks do not offer an API endpoint for receiving and modifying social network notifications. Nevertheless, we used the mockup for explaining the application to the participants of the focus group. A screenshot of an update notification generated by our application can be seen in Figure 8.18. At the top of the notification, the user can see which post has

been liked or commented on, and by which friend, together with the abbreviated content of the post. Below that, the user can find two buttons, allowing them to give either positive ("I liked that the person commented on the post/I liked that he saw the post") or negative feedback ("I did not like that the person commented on the post/I did not want him to see the post"). If the user decides not to give any situational feedback on the update or does not feel either positive or negative, she can just ignore or swipe away the notification.

To find out which consequences (for example for privacy settings) users expect, we conducted a set of focus groups at our institution. Among the four focus groups that we conducted, two had two participants, and the others three participants. The participants were recurited through postings at our university, and had to be active social network users to take part in the study. The participants were students of different discliplines, with aged between 19 and 31 (mean 25.6). The experimenter noted the most important contributions from the discussion directly, and recorded the interview for a detailed analysis later. As usual for a focus group, the interview was structured more like a discussion that is moderated by the experimenter, rather than a Q&A session.

The topics discussed in the interview followed the same structure for all groups:

1. "Has somebody in the group, or one of your friends, ever experienced that you were notified about a new comment or like on one of your posts, and you had a very positive or negative feeling about it, such as being really happy he read and commented on your post, or wishing he had not seen the post at all?"

2. "What did you do in this case? What do you think would be a good reaction?"

3. "If you could tell the social network provider to change something on their webpage, to add a certain functionality or something like that to support you in this case, could you imagine how that could look?'

4. The idea of in-situ feedback on social network update notifications was explained to users without showing the mockup prototype.

5. "How could such a system look? Which feedback options would you like to have?"

6. The smartphone with the update notification (see Figure 8.18) was *shown and explained* to the participants. Participants were told that pressing the "positive" button means "I like that the post was seen and commented on" and "negative" means "I dislike that the post was seen and commented on".

7. "Should the privacy settings be adapted when either positive or negative feedback was clicked? If yes, how?"

8. "Should the privacy settings for the best friends of the affected person also be changed? If yes, how?"

9. The participants were each given an empty sheet of paper and had five minutes to write down further possible applications of the feedback data.

10. After the time was up, participants were requested to discuss the noted ideas with each other.

The experiment ended with a short feedback round about the experiment itself, and whether there were any open questions or things they wished to mention, and then the participants received their payment afterwards. The aforementioned procedure was reviewed and approved by the ethical review board of our instituiton.

8.4.2 Results

All participants were active social network users, using the social network sites on a daily or near daily basis. All of them owned a smartphone that was used on a daily basis; three of them had an additional smartwatch that they used rarely (2 participants) or not at all (1 participant). Six out of eight participants stated they had already experienced a situation where they felt negative about a comment or like and where they would have preferred that the person had not seen the post. All of them stated they experienced at least one situation where they felt positive about a comment. In order to cluster the given answers into several topics, we used an axial coding approach [315].

In the following subsections, we will present the results according to the design questions posed in the introduction.

Current reaction to unwanted comments/likes

The reactions to the discovery of an undesired audience via negative comments and likes were very diverse. We characterized the different types of reactions as follows, denoting the the source of the unwanted comment or like as *the commentator*, and the original poster that gave in-situ feedback on the *commentator's* post as *the user*:

Deletion and reposting to a different audience All ten participants stated they would either delete the post without reposting it (four participants) or publish the same post again to a smaller audience, by using friend lists or changing the disclosement from public to friends only.

Adaptation of friend's privacy settings Three of the participants stated that such a behavior would also directly affect the disclosement settings for the commentator. Depending on whether she was a close friend or not, she would either be blocked on Facebook, or put on the *restricted* friends list, so that she would not see future posts.

Adaptation of general privacy settings Another three participants stated that such a case would make them question their trust in their general privacy settings, so they would take their time to review all of their settings, rearrange friend groups and adapt the privacy settings accordingly so that such a case would not happen again in the future.

When questioned about the effort needed for the measures mentioned, all participants stated that they are very effortful. One of the groups mentioned that the measures they are currently taking are too slow to prevent further damage, so a new functionality should be included in social networks to assist in such situations.

Possible new functionalities to overcome the problem of negative updates

Before presenting our new interface, we asked the participants whether they could think of possible solutions for how social network sites could prevent such a situation or assist them when it occurred. The results can be clustered as follows:

Core group One of the groups came up with the idea of publishing the post to a *core group* of the closest friends first, so they could check and review the post. If one of the friends gives a positive review (for example by pressing a "positive" button on the facebook app) or if there was no negative review after a certain amount of time, the post is published to the remaining friends.

Direct discussion with the commentator One of the participants mentioned that for her, just removing the post or blocking the person is not a complete solution, as the problem with the commentator is not solved. She therefore proposed that a functionality should be implemented which allows the user to directly contact the commentator to discuss why he posted that comment or like, so that the problem is solved directly, and future conflicts can be avoided.

Delayed posting Another group mentioned that removing a person immediately from the post audience could be seen as rude by the commentator. They therefore suggested to add a functionality where the commentator could be removed from the audience after a certain delay, like a week, so the commentator is removed from current and future posts without noticing, and he will not feel offended.

Automatic audience selection by tie strength Using an automatic or semi-automatic selection of the post audience based on tie strength was proposed by some of the participants. The tie strength denotes the *social interconnectedness* between the user and her friends, based on social network data like frequency of chat messages, profile visits, number of likes or comments, etc. The participants proposed two different approaches. One would use the topic of the post and the friend group to automatically determine whether the person should see the post or not (as described in Chapter 4); the other suggests a user interface to manually do the task. The participants described a user interface where the friend images are shown with ascending tie strength, so that the user can draw a line at a certain level of tie strength, above which the post should be published.

Direct or daily feedback on update notifications
Direct feedback was also suggested by two of the groups. In the first discussed option, the participants proposed to add some buttons to the update notification, which allow them to directly block and notify the commentator, to change the privacy settings for the post, and/or hide future posts from the commentator. The second discussed option saw this process more as a daily review process, for example at the end of the day. For this purpose, the social network provider offers a "notification dashboard" where the user can see all update notifications together. She can then scroll through the updates and mark the updates as positive/helpful or negative. For the case of negative updates, the privacy settings are directly applied so that the commentator cannot see the post.

Presentation of the interface and discussion of desired consequences for privacy settings

After discussing possible solutions, the participants were shown the notification message with the feedback opportunity, as shown in Figure 8.18. As stated in the methodology section, we first asked which direct consequences are expected when one of the feedback buttons is pressed, e.g. how privacy settings of the commentator and his closest friends should be adapted. We received the following answers:

Direct impact on the commentator When asked for the direct impact on the disclosure settings for the commentator, we received answers with a widely differing impact on the commentator's disclosement setting:

- In the weakest form, one participant proposed a discussion approach, where the comment is deleted and a chat system is opened to discuss the comment with the commentator. Only if the discussion is not succesful, the commentator is then put on a "restricted" list.
- Two more participants would block access to the commented-on post only.
- Hiding future posts from the commentator's timeline only is the desired solution for two other participants.
- Four participants would record the amount of negative feedback and put the commentator on the "restricted" list at some point. One would do it after the first, one after the second, and two others only after additional cases of negative feedback, depending on whether it is a close friend or not.
- Finally, one person wished to block a person completely from Facebook after the first negative feedback.

Impact on the direct friends of the commentator The negative feedback had no influence on the closest friends or the friend group of the commentator for most of the participants (6 out of 10). However, two subjects would also adapt the privacy settings so that the close friends could not see the post either, based on the tie strength between user and commentator (one subject) or the topic of the post (one subject). An additional two subjects would like to be asked by the software whether the group should be removed from the post, so they could choose on their own.

Further applications of the user feedback

We then collected further ideas by first letting participants write down ideas on a sheet of paper and then discuss the collected ideas with each other. Two groups came up with a similar idea: The feedback could be continously collected to create some kind of "positivity ranking" for all social network friends: each instance of positive feedback adds one point for the commentator, while each negative one removes two or more points from the commentator's positivity score. Using this score, the friends can be sorted by descending "positivity". The applications of such a score would be versatile:

Privacy settings The score can be used to decide whether a post should be displayed to a friend's newsfeed, whether it should be accessible only if the friend visits the user's personal profile, or whether it should be not visible at all.

There should be two different threshold levels that split the user's friends into the three above-mentioned groups and that can be adjusted by the user, in the best case by using a user interface with profile pictures and names of the users and a line which can be moved using drag & drop. Friends that are above the first line will see the user's posts on their wall, friends between the first and second line only if they visit the user's personal page, and friends below the second line not at all. One participant stated that the user should be notified if one of his friends is falling below or rising above one of the thresholds, so the user can review and decide whether the privacy settings should be adapted as proposed.

Ranking newsfeed Another useful option for the positivity ranking is the ranking of the social network newsfeed according to the positivity score in addition to the time the post was published. Posts of friends with a higher score are placed earlier, whereas those of friends with lower scores are placed further down the page or not displayed at all. Also here, a user interface to define the cut-off threshold was favored by the participants.

Content filtering and emphasis Some social networks like Facebook tend to automatically generate "friendship movies", for example when two persons have been friends for one year, or one year after they posted an event together. The positivity score could also be used to decide which friends should receive such a friend video, either to reward friends for their friendship when they have a high positivity score, or on the other hand, to cheer up friends with a negative score so the friendship can again be stronger in the future. The user should also be notified more about life events of friends with a high positivity score, for example their birthdays. On the other hand, persons with a very negative score should receive fewer status updates and posts about the user.

Friendship dashboard Some subjects stated they would like to have a positivity dashboard, where they can see all their friends together with their positivity score, details about their positive and negative actions, and the option to review their privacy settings and friendships. That way, users can review and adapt their privacy settings manually and unfriend or block friends or put them on a restricted list, so that they will not see any future posts. That way, situations like those discussed at the beginning of the experiment would become more unlikely.

Problem discussion Another functionality that could also be integrated into the dashboard is a discussion functionality with friends with a low positivity score. That way, the user could discuss the possible problems that may have led to the negative comments and clarify them, so that they will not happen again in the future. Nevertheless, if the discussion fails, there should also be a button to block the person or put her onto a "restricted" list.

Lastly, the functionality could also be used to give feedback on friends' original posts. That way friends can easily notify the original poster when he is about to do something without due consideration, so he can delete the post quickly in order to avert further damage.

8.4.3 Discussion

Direct adaptation of privacy settings not possible

Unfortunately, as our results have shown, there is no general rule on how the feedback should affect the user's privacy settings, neither for the person that wrote the feedback, nor for the friends or the friend groups she is part of. According to the results, there is also no lowest common denominator that can be used as a minimal solution: whereas some of the users expected the social network to open a direct communication with the friend to discuss what information he saw and that he wasn't supposed to see that post, so he does not further spread the information, others would put the person on a list of restricted users directly or after a certain amount of negative feedback, whereas others would block the user instantly. This also holds for the direct friends of the user or the friend groups containing the friend. The majority do not expect any influence on the closest friends of the user, whereas four out of ten users want them to be put on a restricted list after approval of the user. Therefore, we see little value for using in-situ feedback on privacy violations for adapting privacy settings, neither based on a rule set derived within a large-scale user study, nor based on machine learning techniques using a large data set.

Privacy settings through positivity ranking

Nevertheless, the feedback can be used for creating a positivity rating by summing up positive and negative ratings for a user, that can used for an indirect adaptation of privacy settings through a user feedback cycle, as described in the last section. Based on the friendship score, the friends are separated into three friend groups: The topmost group ("green list") receives the user's posts and updates directly on their news wall. The next group of friends ("yellow list") also has full access to the user's updates, but will see them only on the user's personal page and not in their news feed. Finally, the last group of friends ("red list") does not receive any of the user's posts. The user should have an interface to define the boundaries between the red, yellow and green lists and should be notified when a person is about to move to another list based on the user's feedback.

There have been several approaches that use tie strength between the user and his social network friends instead of the positivity ranking to limit the audience for social network posts [180]. Whereas the approach by Kauer et al. [180] relies on a slider approach to define the cut-off between friends that should receive and friends that should not receive a post, OmniWedges (see last section) uses a radar metaphor to do so. Studies have shown that PrivacyWedges significantly reduces the amount of false positives, i.e. the amount of friends that received the post even though the user did not intend them to. In contrast to the positivity ranking, both approaches rely on the tie strength measure reported by Facebook, which is calculated indirectly using measures like the frequency of private messages, comments, profile visits, etc. Whether the same or better results can also be achieved with the positivity rating should be elaborated upon in future research.

Granularity of the feedback mechanism

As space is very limited on smartphone devices, we concentrated on a minimalistic design for the update notification messages on our smartphone application, offering only a binary feedback choice (positive and negative). However, it might be possible that a more fine-grained feedback option is needed, involving three, four or more

buttons. Users often receive feedback that is on one hand negative, but that is useful for further improvement or that leads to a different, new point of view. Such *helpful feedback* might have a different impact on the positivity rating than destructive negative feedback. The same applies also for positive feedback, which can be helpful or not. We would like to investigate the optimal number of feedback options and the desired effects on the positivity score in future work.

Ethical issues involved in a positivity ranking

Allowing content filtering based on a ranking score is a mechanism that can be seen as censorship, similar to a "social credit score" that is used to rank a country's citizens according to their loyalty to the government. Such a mechanism can distort reality (create a "filter bubble") when negative (but possibly appropriate) comments are hidden from the posts, or when commentators are removed from the audience for future posts. It is therefore highly important that the decision as to whether or not to include a friend in the post audience is one that should not be made automatically by the system, but must be made by the user herself. The social network site can assist the user in offering him a positivity score and a ranking of friends, but it should not be allowed to set the privacy settings automatically. The user still has to think about how he groups his friends together, based on the thresholds that he can adapt, in order to define which users should receive his posts and which should not. In this way, the user can decide whether he wants to use filtering by assigning friends to the different privacy groups and have an "optimized" social network experience without negative emotions, or whether he wants to see the plain hard truth of all positive and negative comments or posts from *all* users.

Other applications of the positivity ranking

In addition to privacy settings, there have been discussed several other ideas that can be applied to the positivity ranking, which can be divided into two different categories: friendship monitoring and content elicitation.

The former could be perfectly combined with the privacy setting approach discussed in the earlier subsection: a new friendship dashboard could be implemented where users can see all of their friends divided into three blocks with an according color, depending on the privacy group they are currently in, together with their positivity score, recent actions and the user's feedback on their actions. In addition to the displayed information, adapting the privacy settings should also be possible within the dashboard: for each user, there should be a functionality to block him or put him on a restricted list. Also, setting the threshold levels could be realized by a draggable slider between the three groups, or by using drag & drop to switch groups for specific friends.

The latter is already partially being done by social network providers. Facebook, for example, displays or hides content based on calculated tie strength. It is also possible for the user to give explicit feedback on some posts, so that "fewer posts like this" are displayed in the future. Nevertheless, studies have shown that the tie strength and friend lists do not completely fit the tie strength order that a user would expect [311]. Using the positivity ranking could therefore lead to better content elicitation than the current standard. Whether this is the case should be investigated in a future study.

8.4.4 Conclusion

In-situ feedback has been proven to be very effective due to its realistic nature and rich context information. It is therefore used in a wide variety of application scenarios. Using in-situ privacy feedback on social network notifications has *not* yet been one of these scenarios. In contrast to traditional scenarios like remote software evaluation, the desired changes in the privacy settings for in-situ privacy feedback are not immediately clear. We held focus groups with a design prototype to capture the desired effects of the user feedback. Although we found that there is no general rule on how the feedback should affect the privacy settings, we have outlined an approach involving a friendship dashboard, which can be used to review the feedback ratings, and for grouping the user's friends into different categories that reflect the privacy settings for future posts for the respective group of users. In addition to managing privacy settings, the dashboard could also be used for content filtering in social networks, which we would like to elaborate upon in future research.

Chapter 9

Conclusion and Outlook

In this final chapter, we will sum up the research topics discussed within this thesis. We will highlight major contributions in the three examined categories, namely privacy recommender systems, privacy user modeling and privacy user interfaces, that support the user in various touchpoints in the privacy journey (see Chapter 1). Nevertheless, the scope of this thesis is limited. Therefore, we will also discuss possibilities for future work that can be conducted based on the insights of the thesis.

9.1 Summary

Privacy is a term that has grown and changed over the years from ancient Roman times to the age of the internet and social media. Whereas the rise of mass media can be seen as the first major threat for data privacy, the media creators in that era were solely professional users, like reporters. In contrast to that, the rise of the internet as a second big wave of privacy threats allows also lay users, for the first time, to create and share media with a large audience on websites and social media, making it significantly harder to control the flow and spreading of information. It is especially a problem that users unintentionally share data with a significantly broader audience than intended. The consequences that can arise from this oversharing are manifold. The data can be used for social engineering attacks, deriving social security numbers needed for legal acts, profiling and stalking. Even governmental organizations and employers use social media for screening potential immigrants and employees. As we have discussed throughout the thesis, the process of choosing privacy settings consists of more steps than merely adjusting the privacy settings. There are preparatory actions needed, like the grouping of recipients or the creation of a user model for a personalized recommendation. Some actions cover the continuous assessment of the settings and capture of privacy violations, that allow the user to refine the chosen settings in the aftermath. We identified several of those actions that we called *touchpoints* in the *privacy journey*. Figure 9.1 shows the approaches that we discussed for the different touchpoints on the privacy journey throughout the thesis.

At the beginning of the journey, we support the user in grouping the recipients, which can later be used for defining the audience of a data item, for example a social network post. Earlier work concentrated on automatically deriving recipient groups based on factors like the recipient's home address, workplace, etc. However, none of the algorithms was able to group the recipients in a meaningful way that was perceived as useful by the users. This led us to research question (RQ) 2, asking how we can motivate a user to do recipient grouping. We followed a new approach trying to make the task of friend grouping more enjoyable by involving new interaction techniques and gamification concepts in the task. Studies on our approach have shown that our design led to a significantly improved user experience, which was stated to be more likely used in everyday social network usage compared to a conventional

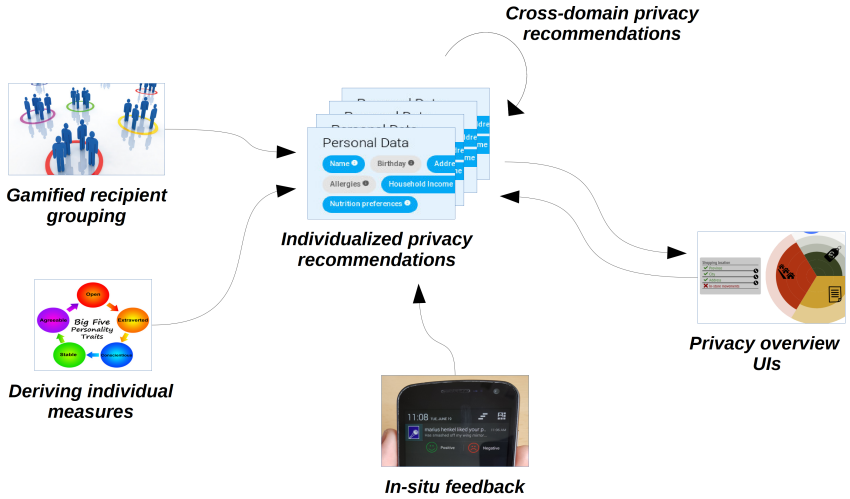


FIGURE 9.1: Approaches supporting several touchpoints on the privacy journey proposed in this thesis.

interface. However, we also found that the error rate is a critical factor that increases with an increased degree of gamification, leading to the conclusion that the gamification of serious tasks is only possible to a certain degree while keeping the quality of the results at the same level.

Similar to the recipient grouping, the privacy user modeling also has to be done before the individualized privacy recommendation. Whereas earlier approaches relied on questionnaires to capture privacy and personality measures, we had the goal to find a way to infer the user model without introducing an additional user burden (RQ 1). We used text analysis software that extracts the frequency of words related to a finite set of word categories, ranging from words about hobbies and religion, over several other categories, to sports topics, and used these measures as an input to predict the privacy and personality measures using machine learning. The results show that, using social network posts of a user, it is possible to infer not only personality, but also the IUIPC privacy measures of a user, whereas the Westin privacy categories cannot be predicted.

Based on these two inputs, we investigated whether the individual factors have a significant impact on the privacy levels, a more general form of privacy settings (RQ 3). Research in the past has concentrated mainly on context factors for the prediction, or used the personality to cluster users and assign them a set of privacy settings that fit their general privacy stereotype. We proved that in the four domains observed within this thesis, using solely individual measures for deriving the privacy levels allows a prediction significantly better than random. This leads to the conclusion that individual measures should be included for a prediction of privacy settings, and are very likely to increase the recommendation quality of recommenders based on context factors. To be more precise, we examined the recommendation of the privacy levels for the recipients of a social network post, the recipients of a shared location, and the permissions for mobile smartphone apps, as well as for data collected inside an intelligent retail store, where sensors record customers' movements, products of interest, products in the shopping basket, etc. We found that the individual measures that should be used as an input for the prediction depend strongly on the domain.

If neither a user model nor access to previous privacy decisions is available, the last possible data source for recommending privacy settings is privacy settings that

the user chose in another domain (for example recommending privacy settings for a shared location based on the user's permission settings on her smartphone). In the scope of this thesis, we conducted a user study examining *which* other domain privacy levels can be used for a prediction and which granularity is needed for best results (e.g. using one general privacy level for the whole domain vs. using a number of privacy levels for a domain depending on context factors). Our studies identified two clusters that allow the prediction of each other's privacy levels. The privacy levels of the mobile app domain allow a prediction of the privacy levels from the intelligent retail domains and vice versa, while the same holds for the social network and location sharing domains. Interestingly, a general privacy level for the whole domain is sufficient as an input in most cases.

Finally, we posed several research questions to be addressed by user interfaces and interaction principles. First, RQ 5 asked how radar interfaces, which have been shown to motivate users in engaging with their privacy settings, can be enhanced to support a large amount of data items, for example recipients of a data item, and how they can be altered to support the display of multiple privacy settings at once (for example for multiple groups of recipients). We designed a radar interface that uses several design elements to be able to display hundreds and even thousands of recipients, the social network friends of a user in our case. In a study involving the users' real social network profile and friends, we were able to show that our new design significantly reduces the amount of errors, namely unwanted disclosures and missing recipients. Furthermore, we proposed a design that renders a three-dimensional *privacy pyramid* out of multiple privacy settings for multiple recipients. Our study has shown that the privacy pyramid is superior for detecting unusual and potentially erroneous settings, allowing the user to review and adapt the critical settings in a conventional list-based interface. Second, we examined how we can capture privacy violations as they occur using a mobile phone app; we were interested especially in consequences for the privacy settings that users expect when giving in-situ feedback (RQ6). Although we found no general rule on how privacy settings should be adapted in our focus group studies, we were able to develop an idea for how the feedback can be accumulated to create a friendship score, which can be used within a user interface ("friendship dashboard") to support the user in assigning disclosement settings and adjusting content filtering settings based on the friendship quality.

The next section will highlight the contributions that the thesis achieves within each of the mentioned research questions.

9.2 Contributions

To summarize, our work provides the following contributions to the reseach field:

1. **Usage of personality and privacy measures for individualized privacy recommendations using multiple privacy levels (RQ 3, Chapters 4-6)**: Our approaches use a user model based on personality and privacy measures to predict individual privacy settings tailored to the user's individual measures rather than using the same privacy settings for each user assigned to the same stereotype. Furthermore, we are the first to predict privacy levels that can be transformed into more detailed privacy settings, allowing users for example, to share only the current city with specific users, instead of the detailed GPS location. The results show that in the four examined domains, individual factors play a significant role in the privacy decision and thus should be included as an additional factor for the prediction together with other factors, like context factors. We have shown that using only individual factors, it is possible to predict privacy settings with a precision significantly better than random. Furthermore, we have shown that the individual measures that have the highest impact on the privacy levels are dependent on the domain. For one of the observed domains, personality measures did not have an effect on the privacy decision, according to our results.

2. **Derivation of privacy measures from written text (RQ 1, Chapter 3)**: We replicate existing research on deriving personality measures from written text and are the first to examine the correlation between the frequency of words per word category and the privacy measures of a user. The study results show that it is possible to derive personality and also the IUIPC privacy measures with our regression-based approach.

3. **Usage of privacy levels from other domains for a recommendation (RQ 4, Chapter 7)**: Within this thesis, we investigated whether privacy recommendations can also be derived using the privacy levels from other domains. We identified two clusters within the four observed domains, which allow us to predict the privacy levels of one domain using the privacy levels of the other domain. The study results show that a prediction is already possible with a single average privacy level of the other domain in the cluster (MGR and MCR method). Having multiple context-dependent privacy levels of the other domain as an input (CGR and CCR method) can increase the prediction precision only in some rare cases.

4. **Radar interfaces able to display a large amount of data items and multiple privacy policies at once (RQ 5, Chapters 8.2 and 8.3)**: The thesis discusses a design that increases the amount of items displayable in the radar. According to our study results, this design allows an audience selection with a large number of recipients with a significantly decreased error rate compared to conventional list-based UIs. Furthermore, we propose a design allowing us to display multiple privacy policies in a three-dimensional privacy pyramid, which allows users to detect potential misconfigurations more easily than with a conventional interface, according to the study results.

5. **Approach to engage users in sorting tasks by optimizing both usability and enjoyability (RQ 2, Chapter 8.1)**: Our results have shown that increasing the user experience motivates the user in doing the task and makes it more likely

that users perform the task as a part of their daily routine. The interaction time does not have a significant influence on the motivation. However, our results also indicate that increasing the degree of gamification also increases the error rate. Therefore, one should always consider the a trade-off between motivation through gamification and quality of the results affected by the increased error rate through gamification.

6. **Approach for capturing privacy violations as they occur and using them to improve privacy policies (RQ6, 8.4)**: The studies described within this thesis have shown that it is not possible to infer a direct rule for how the reporting of privacy violations should affect the privacy settings. Still, we were able to develop design ideas with the participants of the study, leading to the concept of a privacy dashboard displaying potential recipients according to reported feedback, allowing the user to manage her privacy settings based on the given in-situ feedback.

Although the thesis offers a number of contributions, there are still open questions and starting points for future research, which will be discussed in the next section.

9.3 Future work and limitations

Throughout the thesis, we discussed approaches that cover several touchpoints in the privacy journey. In order to reduce random effects, the approaches were tested independently from each other. However, it would be interesting to understand how the recommendation of the complete privacy framework are perceived by users. In particular, using the results of an automatically derived user model for recommending privacy settings in productive environments of the examined domains, for example in cooperation with an established social network provider, would be of interest. As we have seen, both components infer a certain prediction error; therefore, future studies should determine the error arising when both components are combined, and especially whether the result of such a system is still perceived as useful by the users. Apart from this, also the acceptance of the privacy framework as a whole, including the user modeling tool, privacy recommendations, UIs for reviewing and adapting the settings and usage of in-situ feedback in a productive environment, would be one of the major long-term goals for future work in cooperation with a social network provider.

The studies included in this thesis gathered data from (online) questionnaires and are sufficient for a machine learning analysis, and to find out *whether* the individual factors and a personalized privacy setting recommendation is a fruitful approach, which leads to an increased prediction precision if used for deriving the permission settings. However, the study cannot show *how* precise the approach can be if a large permission database is used, as has been done for other approaches based on context factors without personalization. However, machine learning algorithms typically gain in prediction precision with an increased amount of training data. The results shown here can therefore be seen as a lower bound for the precision that can be achieved with a large data set, such as what social network providers like Facebook could use for training such a system. Further research on how much the precision can be increased with large data sets is therefore one further step that could be taken within a research collaboration with a social network provider. In this thesis, we used a generic user model, including personality and privacy measures

for predicting the privacy settings or privacy levels in three of the four examined domains. Only for the social network domain, we added additional domain-specific questions to our set of generic privacy and personality questions to increase the prediction precision. Although generic questionnaires have the advantage that they can be used in several domains, the precision can be further increased if additional domain-specific questions are introduced (see Chapter 4). We would therefore like to examine in future work whether and how much the recommendation can be improved by using domain-specific questionnaires and thereby specialized user models. Within this case, it is especially interesting whether domain-specific measures of one domain can be used to infer domain-specific measures of another domain, similar to the cross-domain privacy setting prediction described in Chapter 7.

Also the privacy levels and their implementation are only described on a theoretical level at the moment. As all social network providers shut down almost all functionalities of their API for public use, it was not possible to implement the described privacy levels as a social network app or plugin. Although we evaluated the acceptance and usefulness of the given implementations of the privacy levels in the user studies presented in this thesis, we would like to conduct a bottom-up design process in future work, also involving end-users in generating ideas on which implementation of privacy levels are reasonable, and then to validate the choice of these design thinking sessions in validation studies later. The design of the privacy dashboard for the in-situ feedback was only discussed in theory, without having any prototype of the final UI at hand. In future work, we would like to design such a privacy dashboard, and evaluate especially its utility for managing privacy settings based on the user feedback.

For the social network and location sharing privacy setting recommenders, we concentrated on a fixed number of topics and occasions as a context factor used as an input for the ML prediction. Although we conducted a pre-study in order to provide post topics which are very common and frequently used, we are aware that the set of topics and occasions does not form an exhaustive list. Therefore, we had to find a compromise between practicability and degree of realism in our validation study. It is not possible to implement a trained system that covers *all* possible individual topics or occasions and friend groups for each participant. A fully individualized system would operate better, but the results of the validation section have shown that precision and user acceptance is already high with a non-individualized system.

A manual annotation of the topics of a user's posts or the occasions of the shared locations, as described for the social network and location sharing recommenders, would not scale to handle a real social network user profile, where each user has up to hundreds of posts and shared locations on his personal profile. To maintain scalability of the approach, a final version would use machine learning to cluster and label a user's posts. In a first step, text cluserting libraries like *carrot*^{2} [241] cluster the existing posts of a user into groups of topics. Each of the topic groups is assigned a topic label by the clustering engine. In the second and final step, the user checks the topic labeling of the groups, and edits incorrectly labeled clusters. Whenever a new post is created, the post is assigned a topic label, based on its assignment to one of the earlier created topic clusters. In this way, besides the initial questionnaire which is filled out, only a little user input is needed to setup and use the system: post topics and privacy settings are assigned automatically by the two machine learning components. Only if either the topic classification or the privacy setting derivation is incorrect, additional user input is needed. Similarly, the occasions could also be derived by clustering shared locations together based on their distance and labeling them.

Apart from that, it is also possible that a post or shared location belongs to more than one topic or occasion, or is at the boundary between them. Although a clustering approach like the former will assign the post to exactly one cluster, the prediction mechanism can also handle a post that is tagged as belonging to more than one topic or occasion. In this case, the prediction will give us several privacy policies, one for each topic. A merging algorithm would go through all of the user groups in each of the policies, and according to Ravichandran et al. [272], use for each observed group the according privacy level depending on the conservativeness ratio of the user in the merged policy. The same merging technique can be used for friends which appear in multiple friend groups. Although this merging is not yet implemented and evaluated, it will allow us to predict posts with an arbitrary number of topics in a future version.

For the derivation of individual measures, we focused on the two currently most popular social networks, namely Facebook and Twitter, to record written text for the study and to perform the prediction. Nevertheless, there are several other social media websites that are getting more and more important recently, and that are often not used in research. To name only one example, we did not examine Youtube comments and posts as a source for the prediction. As related work has shown, a rough prediction of personality features is also possible using image features like brightness, hue, or different color values [112]. Maybe the content of YouTube videos can also be used for such a prediction. We would also like to explore other data sources apart from written text on different websites, including image or video content, to predict our set of privacy and personality measures.

In the cross-domain privacy setting prediction, we were able to identify several coefficients for the context-based regression methods (MCR and CCR) that can be of use in future work. We found that the four domains treated in this chapter can be clustered into two clusters, inside which a regression of the partner's privacy levels is possible. However, there are plenty of other domains that have not been part of our research so far. In future work, we therefore want to investigate whether other domains also form clusters together, or whether they are part of one of the two aforementioned clusters. The ultimate goal is to find out whether there is a finite number of clusters that allow a prediction of each other's privacy levels, what domains they include, and which common properties they share that make them belong to the same cluster. Apart from that, we investigated only a finite amount of context factors in this chapter that have been found to be significant in related work. There might be still other context factors that have not been discovered yet, which we would propose as topics for future work.

We found differences in the prediction precision between the methods using the mean domain privacy level as an input (MGR and MCR) and the methods using context-based privacy levels (CGR and CCR). In some domains, the CGR and the CCR outperform the MGR and MCR. However, the difference is below 10% and therefore relatively small. In future work, we would therefore like to perform a field study, where users have to use a social media account, for example, including privacy settings based on one of the aforementioned approaches, and the task to use the account for some weeks and to adapt the privacy settings, if needed. At the end of the study, we will compare the changes to the privacy settings and thereby the number of errors made with each method. Using a questionnaire, we will evaluate the subjective differences on the perceived prediction precision.

We created two different designs for the social network friend grouping task, one trying to improve the usability of the sorting using VR metaphors, and one geared

towards making it a more fun and entertaining experience. Based on different criteria like the possibility to interrupt the sorting task at any given time, or using gamification elements to enhance the user experience, we came up with two different designs for our user study. Notwithstanding, plenty of other possible design ideas exist and might be suitable for this kind of task. Still, we were able to show that, with our design ideas, the user experience as well as the perceived usability could be improved. However, we would like to elaborate on other designs in the future, especially game designs that might be more prone to errors than our can knockdown game, although the increased error rate might be an effect of the gamified design, which would hold for other game types as well.

In the validation of the friend grouping interfaces, we used the current standard interface as a baseline in order to minimize side effects and to get a comparison of our designs to the current working standard. Although we were able to prove that both usability and user experience were higher using the VR design, we would like to elaborate more on the parts of the design that lead to this effect. We would especially like to discover which of the developed metaphors used in the pragmatic and the playful design led to an increased rating, whether it was the representation of the friends as friend frames inside a shelf, the friend boxes, or the interaction by inserting the frames inside the box. Also, the usage of virtual reality alone might already lead to some effect, at least in terms of user experience. In several follow-up studies we would like to find out more about which design elements have a positive effect in VR using A/B testing, and give concrete guidelines on which metaphors should be used and which should be avoided.

Also for the radar-based UI for selecting the post audience, we had to make compromises at some points in our experiment to increase realism as much as possible while reducing the amount of side effects. The largest compromise we had to make was the grouping of friends and the friend order according to tie strength. Friend groups are already offered automatically by Facebook, but they often contain incorrect information, i.e. friends that should not be in the list, or friends that are missing. The same holds for the tie strength calculated by Facebook based on the frequency of interactions like chat messages, likes or profile visits. For the sake of this lab study, the goal was to judge the user interface and its functionalities without the influence of other side effects like the correctness of these ranking and grouping mechanisms. We therefore let the users manually create their friend groups and correct the tie strength ordering in the experiment. However, we are aware that a typical user will not perform these tasks in her daily social network use, especially as tie strength changes often and has to be adapted accordingly. As a first step in future work, we would therefore like to integrate our approach into a social network website and evaluate the usage frequency of our tool against the standard audience selection functionality, especially when using the already existing friend groups (created either automatically or by the user) and the tie strength calculation offered by the social network provider. The same holds for the hypothetical posts for which users had to select their audience during the study. In order to allow a fair comparison and to reduce side effects, we had to show the same set of posts to each user. Therefore, we could not use the user's actual social network posts, as these might lead to different results because of different sensitivity levels of the posts. Supported by the results from the current study, we would like to conduct an in-the-wild study with a UI integrated into a social network site as a next step, with a focus on evaluating the usage ratio and user acceptance of our UI rather than comparing it to the current standard in a controlled lab study

In the in-situ feedback approach, we conducted only four focus groups to get a

first impression on whether the update notification feedback can be used to adapt the privacy settings based on a rule set, and which alternatives may arise. The discussions clearly show that the former is not the case. Nevertheless, the discussions led to an outline for how an alternative could look: first, feedback data is collected in order to form a positivity score for all friends. Using a friendship dashboard, the user can easily get an overview on the current state of the positivity scores, and define his privacy settings based on three different groups depending on the recent scores. He can adapt the threshold levels that define the boundaries between the different groups, and adapt the privacy settings by that means. In this way, we would like to explore whether the outlined idea is accepted by users using an interface prototype. If the dashboard is accepted, we will extend the functionalities to allow content and notification elicitation based on the positivity scores. Although the discussions have already led to a promising idea for a possible approach, further discussion groups could reveal more insightful ideas and alternative directions that could be taken into consideration, if the outlined approach is not as succesful as expected.

Finally, in cooperation with a large social network provider, a retailer like Amazon and a smartphone operating system developer like Google, we would like to evaluate the framework with all its components in a large-scale in-the-wild study, that includes all discussed domains and approaches presented in this thesis. Users should use the components of the framework on a daily basis with their own data and as part of their daily routine, so we can get an insight into whether the framework is accepted as a whole, which components are perceived to be most useful, and which need further improvement to be accepted by users. Qualitative feedback in particular can be of great value for identifying possible problems with the components of the framework.

9.4 Concluding remarks

We presented a framework here that targets the *privacy threat of oversharing data by the user*. There are many other threats that can harm users' privacy in the domains which are not addressed in this thesis (see Chapter 2); therefore, the framework discussed here can only protect a user's privacy if it is combined with other approaches targeting the remaining privacy threats that are out of scope of the thesis. However, developers and researchers can only minimize the privacy and security risks to a certain degree, by assisting the user in privacy tasks or by designing products (either hardware or software) to be privacy preserving by nature (*"privacy by design"*). But in the end, the users themselves are often the weakest link in the chain, as they are often not aware enough of privacy issues so far. Furthermore, new privacy and security threats and hacking attacks are discovered every day; some are kept private for a long time until they become public (*"zero day exploits"*). For those two reasons, this thesis, even in combination with the best approaches for other privacy threats, can only *reduce* the privacy threats in a digital world. A residual risk always remains.

Appendix A

Appendix

A.1 Questionnaires

question	topic number										
	1	2	3	4	5	6	7	8	9	10	all
1. My social network is part of my everyday activity.		X	X	X				X			
2. I feel out of touch when I haven t logged onto my social network for a while.	X	X	X	X	X	X	X	X	X		X
3. On my social network, I feel close to the people in my friend list.				X						X	
4. On my social network, I am updated about my friends.					X						
5. There are several people on my social network I trust to solve my problems.	X	X					X	X	X		X
6. I do not want to post very intimate things about myself on my social network.			X				X			X	
7. I post very intimate things about myself on my social network.	X	X			X	X	X	X	X		X
8. I want to share only minimal information about myself on my social network.									X		
9. I want to be able to choose what to share and what to hold back on my social network.			X	X	X				X	X	X
10. My friends keep personal information they know about me between us.	X		X	X		X	X			X	X
11. I want to limit what personal information my friends share about me on my social network.			X								
12. I want my social network friends to keep personal information they know about me between us.				X	X					X	
13. I only have people in my social network who I associate with on a regular basis in real life.		X	X		X		X		X		X
14. I do not have social network friends who are no longer real friends.									X		
15. I make a distinction between my friends based on the type of relationship I have with them. For example, family, friends, co-workers, etc.		X		X	X		X			X	
16. I manage everything that shows up on my timeline/wall for others to see.			X			X		X		X	
17. I want to approve all content before it is posted to my my social network timeline/wall.			X							X	
18. I want to control who sees the status updates I post.				X						X	
19. I know whether friends of friends can see my posts.	X		X			X	X				
20. I am concerned about who sees the status updates I post.		X		X		X			X		X

TABLE A.1: Questions selected for the social network privacy setting prediction prediction of each topic.

question	topic number										
	1	2	3	4	5	6	7	8	9	10	all
21. I want to restrict others in my network from being able to see who I am and am not friends with on my social network.				X	X						
22. It is not important for me that I am aware of and that I know who can see my personal information.				X							
23. I want to avoid letting specific groups of friends interact with each other on my social network.		X			X						
24. I want to keep my different social circles separate from each other on my social network.	X	X	X	X	X		X		X	X	X
25. I want to moderate how my different groups of friends interact with one another on my my social network page.	X						X				
26. Users' control over their personal information in a social network lies at the heart of user privacy.				X							
27. I think that the privacy in my social network is violated if control over private information is lost or reduced against my will.			X							X	
28. I check my my social network privacy settings regularly.	X				X		X			X	
29. I am satisfied with the privacy settings that my social network offers to the users.							X				X
30. It is very important for me that I am aware of, and that I know who can see my personal information.	X					X		X			
31. It bothers me if other network users ask me for personal information.				X	X				X		
32. If someone asks me for personal information I usually think twice before I give it away.			X				X			X	
33. It bothers me that so many members of the social network are able to look into my personal information.					X				X		
34. I am concerned that other users collect too much information about me.	X				X					X	
35. Consumers have lost all control over how personal information is collected and used by companies.	X										
36. Existing laws and organizaitional practices provide a reasonable level of protection for consumer privacy today.							X			X	X

TABLE A.2: Questions selected for the social network privacy setting prediction prediction of each topic continued.

Bibliography

[1] Fabian Abel et al. “Cross-system user modeling and personalization on the Social Web”. In: *User Modeling and User-Adapted Interaction* 23.2 (2013). Springer, pp. 169–209.

[2] Gunes Acar et al. “The Web Never Forgets: Persistent Tracking Mechanisms in the Wild”. In: *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*. CCS ’14. ACM, 2014, pp. 674–689.

[3] Alessandro Acquisti and Ralph Gross. “Imagined Communities: Awareness, Information Sharing, and Privacy on the Facebook”. In: *Privacy Enhancing Technologies*. Springer Berlin Heidelberg, 2006, pp. 36–58.

[4] Alessandro Acquisti, Ralph Gross, and Fred Stutzman. “Face Recognition and Privacy in the Age of Augmented Reality”. In: *Journal of Privacy and Confidentiality* 6 (2014). Labor Dynamics Institute.

[5] Gediminas Adomavicius and Alexander Tuzhilin. “Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions”. In: *IEEE Transactions on Knowledge and Data Engineering* 17.6 (2005). IEEE, pp. 734–749.

[6] Seyed Hossein Ahmadinejad and Philip W.L. Fong. “On the Feasibility of Inference Attacks by Third-party Extensions to Social Network Systems”. In: *Proceedings of the 8th ACM SIGSAC Symposium on Information, Computer and Communications Security*. ASIA CCS ’13. ACM, 2013, pp. 161–166.

[7] Gordon W. Allport, Henry S. Odbert, and Harvard Psychological Laboratory. *Trait-names: A Psycho-lexical Study*. Vol. 47. Psychological Review Publications 1. Psychological Review Company, 1936.

[8] Mishari Almishari, Ekin Oguz, and Gene Tsudik. “Fighting Authorship Linkability with Crowdsourcing”. In: *Proceedings of the Second ACM Conference on Online Social Networks*. COSN ’14. ACM, 2014, pp. 69–82.

[9] Hazim Almuhimedi et al. “Your Location Has Been Shared 5,398 Times!: A Field Study on Mobile App Privacy Nudging”. In: *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. CHI ’15. ACM, 2015, pp. 787–796.

[10] Irwin Altman. *The environment and social behavior: privacy, personal space, territory, crowding*. Brooks/Cole Pub. Co., 1975.

[11] Baris Altop, Mehmet Ercan Nergiz, and Yücel Saygin. “A Probabilistic Inference Attack on Suppressed Social Networks”. In: *IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*. IEEE, 2012, pp. 726–727.

[12] Mahdi N. Al-Ameen and Matthew Wright. “Persea: A Sybil-resistant Social DHT”. In: *Proceedings of the Third ACM Conference on Data and Application Security and Privacy*. CODASPY ’13. ACM, 2013, pp. 169–172.

[13] Saleema Amershi, James Fogarty, and Daniel Weld. “Regroup: Interactive Machine Learning for On-demand Group Creation in Social Networks”. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI ’12. ACM, 2012, pp. 21–30.

[14] Odin W. Anderson. “Professional Dominance: The Social Structure of Medical Care. Eliot Freidson”. In: *American Journal of Sociology* 77.3 (1971). University of Chicago Press, pp. 599–601.

[15] Mohd Anwar et al. “Visualizing Privacy Implications of Access Control Policies in Social Network Systems”. In: *Data Privacy Management and Autonomous Spontaneous Security*. Springer, 2010, pp. 106–120.

[16] Oshrat Ayalon and Eran Toch. “Retrospective Privacy: Managing Longitudinal Privacy in Online Social Networks”. In: *Proceedings of the Ninth Symposium on Usable Privacy and Security*. SOUPS ’13. ACM, 2013, 4:1–4:13.

[17] Filip Babic. “Rethinking Online Privacy Litigation as Google Expands Use of Tracking: Giving Meaning to Our Online Browsing and the Federal Wiretap Act”. In: *Hastings Communications and Entertainment Law Journal*. Vol. 36. University of California, 2014.

[18] Michael A. Babyak. “What you see may not be what you get: a brief, nontechnical introduction to overfitting in regression-type models.” In: *Psychosomatic medicine* 66 3 (2004). LWW Journals, pp. 411–21.

[19] Yoram Bachrach et al. “Personality and Patterns of Facebook Usage”. In: *Proceedings of the 4th Annual ACM Web Science Conference*. WebSci ’12. ACM, 2012, pp. 24–32.

[20] Randy Baden et al. “Persona: An Online Social Network with User-defined Privacy”. In: *Proceedings of the ACM SIGCOMM 2009 Conference on Data Communication*. SIGCOMM ’09. ACM, 2009, pp. 135–146.

[21] R. Michael Bagby, Margarita B. Marshall, and Stelios Georgiades. “Dimensional Personality Traits and the Prediction of DSM-IV Personality Disorder Symptom Counts in a Nonclinical Sample”. In: *Journal of Personality Disorders* 19 (2005). Guilford press, pp. 53–67.

[22] Paritosh Bahirat et al. “A Data-Driven Approach to Developing IoT Privacy-Setting Interfaces”. In: *23rd International Conference on Intelligent User Interfaces*. IUI ’18. ACM, 2018, pp. 165–176.

[23] Rebecca Balebako et al. “"Little Brothers Watching You": Raising Awareness of Data Leaks on Smartphones”. In: *Proceedings of the Ninth Symposium on Usable Privacy and Security*. SOUPS ’13. ACM, 2013.

[24] Gaurav Bansal, Fatemeh "Mariam" Zahedi, and David Gefen. “The Impact of Personal Dispositions on Information Sensitivity, Privacy Concern and Trust in Disclosing Health Information Online”. In: *Decision Support Systems* 49.2 (2010). Elsevier, pp. 138–150.

[25] John Bargh and Katelyn McKenna. “The Internet and Social Life”. In: *Annual review of psychology* 55 (2004). Annual Reviews, pp. 573–90.

[26] John Bargh, Katelyn McKenna, and Grainne Fitzsimons. “Can You See the Real Me? Activation and Expression of the "True Self" on the Internet”. In: *Journal of Social Issues* 58 (2002). Wiley, pp. 33–48.

[27] Susan B. Barnes. “A privacy paradox: Social networking in the United States”. In: *First Monday* 11.9 (2006). First Monday Editorial Group.

[28] Murray R. Barrick and Michael K. Mount. "The Big Five Personality Dimensions and job performance: A Meta-analysis". In: *Personnel Psychology* 44 (1991). Wiley, pp. 1–26.

[29] Filipe Beato, Mauro Conti, and Bart Preneel. "Friend in the Middle (FiM): Tackling de-anonymization in social networks". In: *2013 IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*. IEEE, 2013, pp. 279–284.

[30] Michel Begin. "Trust or Dependence? The Patient and the Health Care Professional". In: *Center for Health Economics and Policy Analysis fourth annual health policy conference* (1991).

[31] Michael Benisch et al. "Capturing location-privacy preferences: quantifying accuracy and user-burden tradeoffs". In: *Personal and Ubiquitous Computing* 15.7 (2011). Springer, pp. 679–694.

[32] Stanley I. Benn. "Privacy, Freedom, and Respect for Persons". In: *Privacy*. Vol. XIII. NOMOS. Yearbook of the American Society for Political and Legal Philosophy. Atherton Press, 1971, pp. 1–26.

[33] Shlomo Berkovsky, Tsvi Kuflik, and Francesco Ricci. "Mediation of user models for enhanced personalization in recommender systems". In: *User Modeling and User-Adapted Interaction* 18.3 (2008). Springer, pp. 245–286.

[34] Michael S. Bernstein et al. "Quantifying the Invisible Audience in Social Networks". In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '13. ACM, 2013, pp. 21–30.

[35] Jon D. Bible and Darien A. McWhirter. *Privacy as a Constitutional Right: Sex, Drugs, and the Right to Life*. Quorum Books, 1992.

[36] Leyla Bilge et al. "All Your Contacts Are Belong to Us: Automated Identity Theft Attacks on Social Networks". In: *Proceedings of the 18th International Conference on World Wide Web*. WWW '09. ACM, 2009, pp. 551–560.

[37] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer, 2006.

[38] Christian Bizer, Tom Heath, and Tim Berners-Lee. "Linked Data - The Story So Far". In: *International Journal on Semantic Web and Information Systems* 5.3 (2009). IGI Global, pp. 1–22.

[39] Jack Block. "A Contrarian View of the Five-Factor Approach to Personality Description". In: *Psychological Bulletin* 117 (2 1995). APA Publishing, pp. 187–215.

[40] Vincent D Blondel et al. "Fast unfolding of communities in large networks". In: *Journal of statistical mechanics: theory and experiment* 2008.10 (2008). Institute of Physics, P10008.

[41] Jeffrey Boase et al. "The strength of internet ties". In: *Pew Research Center's Internet & American Life Project* (2006).

[42] Angela M. Bodling and Thomas Martin. "Eysenck Personality Inventory". In: *Encyclopedia of Clinical Neuropsychology*. Springer New York, 2011, pp. 1007–1008.

[43] Sophie Boerman, Sanne Kruikemeier, and Frederik Borgesius. "Online Behavioral Advertising: A Literature Review and Research Agenda". In: *Journal of Advertising* 46 (2017). Taylor & Francis, pp. 363–376.

[44] Sissela Bok. *Secrets: On the Ethics of Concealment and Revelation*. Oxford University Press, 1982.

[45] Peter Borkenau and Fritz Ostendorf. *NEO-Fünf-Faktoren Inventar:(NEO-FFI); nach Costa und McCrae*. Hogrefe, 1993.

[46] Bernhard E. Boser, Isabelle M. Guyon, and Vladimir N. Vapnik. "A Training Algorithm for Optimal Margin Classifiers". In: *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*. COLT '92. ACM, 1992, pp. 144–152.

[47] Y-Lan Boureau, Jean Ponce, and Yann LeCun. "A Theoretical Analysis of Feature Pooling in Visual Recognition". In: *Proceedings of the 27th International Conference on International Conference on Machine Learning*. ICML'10. Omnipress, 2010, pp. 111–118.

[48] Alex Braunstein, Laura Granka, and Jessica Staddon. "Indirect Content Privacy Surveys: Measuring Privacy without Asking about It". In: *Proceedings of the Seventh Symposium on Usable Privacy and Security*. SOUPS '11. ACM, 2011.

[49] Carolyn Brodie et al. "Usable Security and Privacy: A Case Study of Developing Privacy Management Tools". In: *Proceedings of the 2005 Symposium on Usable Privacy and Security*. SOUPS '05. ACM, 2005, pp. 35–43.

[50] Michael W. Browne. "Cross-Validation Methods". In: *Journal of mathematical psychology* 44 (2000). Elsevier, pp. 108–132.

[51] A.J. Bernheim Brush, John Krumm, and James Scott. "Exploring End User Preferences for Location Obfuscation, Location-based Services, and the Value of Location". In: *Proceedings of the 12th ACM International Conference on Ubiquitous Computing*. UbiComp '10. ACM, 2010, pp. 95–104.

[52] Tom Buchanan et al. "Development of measures of online privacy concern and protection for use on the Internet". In: *Journal of the American Society for Information Science and Technology* 58.2 (2007). Wiley, pp. 157–165.

[53] Michael Buhrmester, Tracy Kwang, and Samuel D. Gosling. "Amazon's Mechanical Turk: A new Source of Inexpensive, Yet High-Quality, Data?" In: *Perspectives on Psychological Science* 6.1 (2011). SAGE Journals, pp. 3–5.

[54] Rudy Den Buurman. "User-centred design of smart products". In: *Ergonomics* 40.10 (1997). Taylor and Francis, pp. 1159–1169.

[55] Iván Cantador et al. "Cross-Domain Recommender Systems". In: *Recommender Systems Handbook*. Springer US, 2015, pp. 919–959.

[56] Stuart K. Card. *The Psychology of Human-Computer Interaction*. CRC Press, 2018.

[57] Barbara Carminati, Elena Ferrari, and Andrea Perego. "Enforcing Access Control in Web-based Social Networks". In: *ACM Transactions on Information and System Security* 13 (2009). ACM, 6:1–6:38.

[58] Barbara Carminati, Elena Ferrari, and Andrea Perego. "Rule-Based Access Control for Social Networks". In: *On the Move to Meaningful Internet Systems 2006: OTM 2006 Workshops*. Springer Berlin Heidelberg, 2006, pp. 1734–1744.

[59] Alain Celisse. "Optimal cross-validation in density estimation with the L^2-loss". In: *The Annals of Statistics* 42.5 (2014). Institute of Mathematical Statistics, pp. 1879–1910.

[60] Richard Chalfen. *Snapshot Versions of Life*. Project MUSE. University of Wisconsin Press, 1987.

[61] Barbara S. Chaparro, Veronica D. Hinkle, and Shannon K. Riley. "The Usability of Computerized Card Sorting: A Comparison of Three Applications by Researchers and End Users". In: *Journal of Usability Studies* 4.1 (2008). Usability Professionals' Association, pp. 31–48.

[62] Cathy Charles and Suzanne DeMaio. "Lay Participation in Health Care Decision Making: A Conceptual Framework". In: *Journal of health politics, policy and law* 18 (1993). Duke University Press, pp. 881–904.

[63] Barry Checkoway. *Citizens and Health Care: Participation and Planning for Social Change, Innovative Citizen Participation in Health Planning*. Elsevier, 1981.

[64] Jilin Chen et al. "Making Use of Derived Personality: The Case of Social Media Ad Targeting". In: *International AAAI Conference on Web and Social Media*. AAAI, 2015.

[65] Pern Hui Chia, Yusuke Yamamoto, and N. Asokan. "Is This App Safe?: A Large Scale Study on Application Permissions and Risk Signals". In: *Proceedings of the 21st International Conference on World Wide Web*. WWW '12. ACM, 2012, pp. 311–320.

[66] Nicholas A. Christakis and James H. Fowler. *Connected: The Surprising Power of Our Social Networks and How They Shape Our Lives*. Back Bay Books, 2011.

[67] Delphine Christin, Martin Michalak, and Matthias Hollick. "Raising User Awareness About Privacy Threats in Participatory Sensing Applications Through Graphical Warnings". In: *Proceedings of International Conference on Advances in Mobile Computing & Multimedia*. MoMM '13. ACM, 2013, 445:445–445:454.

[68] Delphine Christin et al. "A Survey on Privacy in Mobile Participatory Sensing Applications". In: *Journal of Systems and Software* 84.11 (2011). Elsevier, pp. 1928–1946.

[69] Delphine Christin et al. "Exploring User Preferences for Privacy Interfaces in Mobile Sensing Applications". In: *Proceedings of the 11th International Conference on Mobile and Ubiquitous Multimedia*. MUM '12. ACM, 2012, 14:1–14:10.

[70] Lillian Clark and Levent Çallı. "Personality types and Facebook advertising: An exploratory study". In: *Journal of Direct, Data and Digital Marketing Practice* 15.4 (2014). Palgrave, pp. 327–336.

[71] C. Robert Cloninger, Thomas R. Przybeck, and Dragan M. Svrakic. "The Tridimensional Personality Questionnaire: U.S. Normative Data". In: *Psychological Reports* 69.3 (1991). SAGE Journals, pp. 1047–1057.

[72] C.Robert Cloninger, Dragan M. Svrakic, and Richard D. Wetzel. *The Temperament and Character Inventory (TCI): A Guide to Its Development and Use*. Center for Psychobiology of Personality, Washington University, 1994.

[73] Kay Connelly, Ashraf Khalil, and Yong Liu. "Do I Do What I Say?: Observed Versus Stated Privacy Preferences". In: *Proceedings of the 11th IFIP TC 13 International Conference on Human-computer Interaction*. INTERACT'07. Springer, 2007, pp. 620–623.

[74] Sunny Consolvo et al. "Location Disclosure to Social Relations: Why, when, & What People Want to Share". In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '05. ACM, 2005, pp. 81–90.

[75] Corinna Cortes and Vladimir Vapnik. "Support-vector networks". In: *Machine Learning* 20.3 (1995). Springer, pp. 273–297.

[76] Paul T. Jr. Costa and Robert R. McCrae. "Age Differences in Personality Structure: a Cluster Analytic Approach1". In: *Journal of Gerontology* 31.5 (1976). Oxford University Press, pp. 564–570.

[77] Paul T. Costa and Robert R. McCrae. *Manual for the NEO personality inventory*. Psychological Assessment Resources, 1985.

[78] Paul T. Costa and Robert R. McCrae. *Revised NEO Personality Inventory (NEO PI-R) and NEO Five-Factor Inventory (NEO-FFI)*. Psychological Assessment Resources, 1992.

[79] Lorrie Faith Cranor, Praveen Guduru, and Manjula Arjula. "User Interfaces for Privacy Agents". In: *ACM Transactions on Compututer-Human Interaction* 13.2 (2006). ACM, pp. 135–178.

[80] Nathan Crilly, James Moultrie, and P. John Clarkson. "Seeing things: consumer response to the visual domain in product design". In: *Design Studies* 25.6 (2004). Elsevier, pp. 547–577.

[81] David Crowe and Wasim A. Al-Hamdani. "Google Privacy: Something for Nothing?" In: *Proceedings of the 2013 on InfoSecCD '13: Information Security Curriculum Development Conference*. InfoSecCD '13. ACM, 2013, pp. 27–32.

[82] Mary J. Culnan. "Consumer awareness of name removal procedures: Implications for direct marketing". In: *Journal of Direct Marketing* 9.2 (1995). Wiley, pp. 10–19.

[83] Ritendra Datta et al. "Image Retrieval: Ideas, Influences, and Trends of the New Age". In: *ACM Computing Surveys* 40.2 (2008). ACM, pp. 1–60.

[84] E. E. David and R. M. Fano. "Some Thoughts about the Social Implications of Accessible Computing". In: *Proceedings of the November 30–December 1, 1965, Fall Joint Computer Conference, Part I*. AFIPS '65 (Fall, part I). ACM, 1965, pp. 243–247.

[85] Ralf De Wolf et al. "The promise of audience transparency. Exploring users' perceptions and behaviors towards visualizations of networked audiences on Facebook". In: *Telematics and Informatics* 32.4 (2015). Elsevier, pp. 890–908.

[86] Ratan Dey, Zubin Jelveh, and Keith Ross. "Facebook users have become much more private: A large-scale study". In: *2012 IEEE International Conference on Pervasive Computing and Communications Workshops*. IEEE, 2012, pp. 346–352.

[87] Tobias Dienlin and Sabine Trepte. "Is the privacy paradox a relic of the past? An in-depth analysis of privacy attitudes and privacy behaviors". In: *European Journal of Social Psychology* 45.3 (2015). Wiley, pp. 285–297.

[88] Tamara Dinev and Paul Hart. "An Extended Privacy Calculus Model for E-Commerce Transactions". In: *Journal on Information Systems Research* 17.1 (2006). INFORMS, pp. 61–80.

[89] Tamara Dinev and Paul Hart. "Internet privacy concerns and their antecedents - measurement validity and a regression model". In: *Behaviour & Information Technology* 23.6 (2004). Taylor & Francis, pp. 413–422.

[90] Tamara Dinev et al. "Internet Users' Privacy Concerns and Beliefs About Government Surveillance: An Exploratory Study of Differences Between Italy and the United States." In: *Handbook of Research on Information Management and the Global Landscape* 14 (2006). IGI Global, pp. 57–93.

[91] Tamara Dinev et al. “Privacy calculus model in e-commerce – a study of Italy and the United States”. In: *European Journal of Information Systems* 15.4 (2006). Palgrave Macmillan, pp. 389–402.

[92] Judith Donath and Danah Boyd. “Public Displays of Connection”. In: *BT Technology Journal* 22.4 (2004). Kluwer Academic Publishers, pp. 71–82.

[93] Harris Drucker et al. “Support Vector Regression Machines”. In: *Advances in Neural Information Processing Systems 9*. MIT Press, 1997, pp. 155–161.

[94] Werner Dubitzky, Martin Granzow, and Daniel P. Berrar. *Fundamentals of Data Mining in Genomics and Proteomics*. Springer, 2006.

[95] Catherine Dwyer, Starr Roxanne Hiltz, and Katia Passerini. “Trust and Privacy Concern Within Social Networking Sites: A Comparison of Facebook and MySpace”. In: *Proceedings of the Thirteenth Americas Conference on Information Systems (AMCIS 2007)*. AIS.

[96] Julia B. Earp et al. “Examining Internet privacy policies within the context of user privacy values”. In: *IEEE Transactions on Engineering Management* 52.2 (2005). IEEE, pp. 227–237.

[97] Judy Edworthy et al. “Linguistic and Location Effects in Compliance with Pesticide Warning Labels for Amateur and Professional Users”. In: *Human Factors* 46.1 (2004). SAGE journals, pp. 11–31.

[98] Nicole B. Ellison, Charles Steinfield, and Cliff Lampe. “The Benefits of Facebook "Friends:" Social Capital and College Students’ Use of Online Social Network Sites”. In: *Journal of Computer-Mediated Communication* 12.4 (2007). Wiley, pp. 1143–1168.

[99] Sarah Elwood and Agnieszka Leszczynski. “Privacy, reconsidered: New representations, data practices, and the geoweb”. In: *Geoforum* 42 (2011). Elsevier, pp. 6–15.

[100] Motahhare Eslami et al. “"I Always Assumed That I Wasn’T Really That Close to [Her]": Reasoning About Invisible Algorithms in News Feeds”. In: *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. CHI ’15. ACM, 2015, pp. 153–162.

[101] Hans J. Eysenck. “General Features of the Model”. In: *A Model for Personality*. Springer Berlin Heidelberg, 1981, pp. 1–37.

[102] Sybil B.G. Eysenck, Hans J. Eysenck, and Paul Barrett. “A revised version of the psychoticism scale”. In: *Personality and Individual Differences* 6.1 (1985). Elsevier, pp. 21–29.

[103] Lujun Fang and Kristen LeFevre. “Privacy Wizards for Social Networking Sites”. In: *Proceedings of the 19th International Conference on World Wide Web*. WWW ’10. ACM, 2010, pp. 351–360.

[104] Golnoosh Farnadi et al. “Computational personality recognition in social media”. In: *User Modeling and User-Adapted Interaction* 26.2 (2016). Springer, pp. 109–142.

[105] Shelly Farnham et al. “Personal Map: Automatically Modeling the User’s Online Social Network”. In: *IFIP TC13 International Conference on Human-Computer Interaction*. INTERACT. 2003.

[106] Kassem Fawaz and Kang G. Shin. “Location Privacy Protection for Smartphone Users”. In: *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*. CCS ’14. ACM, 2014, pp. 239–250.

[107] Adrienne Porter Felt, Kate Greenwood, and David Wagner. "The Effectiveness of Application Permissions". In: *Proceedings of the 2nd USENIX Conference on Web Application Development*. WebApps'11. USENIX Association, 2011, p. 7.

[108] Adrienne Porter Felt et al. "Android Permissions Demystified". In: *Proceedings of the 18th ACM Conference on Computer and Communications Security*. CCS '11. ACM, 2011, pp. 627–638.

[109] Adrienne Porter Felt et al. "Android Permissions: User Attention, Comprehension, and Behavior". In: *Proceedings of the Eighth Symposium on Usable Privacy and Security*. SOUPS '12. ACM, 2012, 3:1–3:14.

[110] Adrienne Felt and David Evans. "Privacy protection for social networking APIs". In: *Proceedings of Web 2.0 Security and Privacy (W2SP)* (2009). IEEE.

[111] Ignacio Fernandez-Tobias et al. "A Generic Semantic-based Framework for Cross-domain Recommendation". In: *Proceedings of the 2Nd International Workshop on Information Heterogeneity and Fusion in Recommender Systems*. HetRec '11. ACM, 2011, pp. 25–32.

[112] Bruce Ferwerda, Markus Schedl, and Marko Tkalcic. "Predicting Personality Traits with Instagram Pictures". In: *Proceedings of the 3rd Workshop on Emotions and Personality in Personalized Systems 2015*. EMPIRE '15. ACM, 2015, pp. 7–10.

[113] Bruce Ferwerda et al. "Personality Traits Predict Music Taxonomy Preferences". In: *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems*. CHI EA '15. ACM, 2015, pp. 2241–2246.

[114] Bob Fields et al. "In Use, In Situ: Extending Field Research Methods". In: *International Journal on Human-Computer Interaction* 22.1 (2007). Taylor & Francis, pp. 1–6.

[115] Casey Fiesler et al. "What (or Who) Is Public?: Privacy Settings and Social Media Content Sharing". In: *Proceedings of the ACM Conference on Computer Supported Cooperative Work and Social Computing*. CSCW '17. ACM, 2017, pp. 567–580.

[116] Josef Fink and Alfred Kobsa. "User Modeling for Personalized City Tours". In: *Artificial Intelligence Review* 18 (2002). Springer, pp. 33–74.

[117] Gerhard Fischer. "User Modeling in Human-Computer Interaction". In: *User Modeling and User-Adapted Interaction* 11 (2000). Springer.

[118] Elizabeth Fokes and Lei Li. "A Survey of Security Vulnerabilities in Social Networking Media: The Case of Facebook". In: *Proceedings of the 3rd Annual Conference on Research in Information Technology*. RIIT '14. ACM, 2014, pp. 57–62.

[119] Osmond K. Fraenkel and Alan Westin. "Privacy and Freedom". In: *The Annals of the American Academy of Political and Social Science* 377.1 (1968). SAGE Journals, pp. 196–197.

[120] Thomas Franke. "P3P — platform for privacy preferences project". In: *Wirtschaftsinformatik* 43.2 (2001). Springer, pp. 197–199.

[121] Arik Friedman, Shlomo Berkovsky, and Mohamed Ali Kaafar. "A differential privacy framework for matrix factorization recommender systems". In: *User Modeling and User-Adapted Interaction* 26.5 (2016). Springer, pp. 425–458.

[122] Filip De Fruyt, Lieve van de Wiele, and Kees van Heeringen. "Cloninger's Psychobiological Model of Temperament and Character and the Five-Factor Model of Personality". In: *Personality and Individual Differences* 29.3 (2000). Elsevier, pp. 441–452.

[123] Markus Funk et al. "Comparing Projected In-situ Feedback at the Manual Assembly Workplace with Impaired Workers". In: *Proceedings of the 8th ACM International Conference on Pervasive Technologies Related to Assistive Environments*. PETRA '15. ACM, 2015, 1:1–1:8.

[124] Bo Gao and Bettina Berendt. "Circles, posts and privacy in egocentric social networks: An exploratory visualization approach". In: *IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*. IEEE, 2013, pp. 792–796.

[125] Seymour Geisser. *Predictive Inference*. Chapman & Hall/CRC Monographs on Statistics & Applied Probability. Taylor & Francis, 1993.

[126] Stuart Geman, Elie Bienenstock, and Rene Doursat. "Neural Networks and the Bias/Variance Dilemma". In: *Neural Computation* 4.1 (1992). MIT press, pp. 1–58.

[127] Lise Getoor and Christopher P. Diehl. "Link Mining: A Survey". In: *SIGKDD Explorations Newsletter* 7.2 (2005). ACM, pp. 3–12.

[128] Kambiz Ghazinour, Stan Matwin, and Marina Sokolova. "Monitoring and Recommending Privacy Settings in Social Networks". In: *Proceedings of the Joint EDBT/ICDT 2013 Workshops*. EDBT '13. ACM, 2013, pp. 164–168.

[129] Martin Gisch, Alexander De Luca, and Markus Blanchebarbe. "The Privacy Badge: A Privacy-Awareness User Interface for Small Devices". In: *Proceedings of the 4th International Conference on Mobile Technology, Applications, and Systems and the 1st International Symposium on Computer Human Interaction in Mobile Technology*. Mobility '07. ACM, 2007, pp. 583–586.

[130] Dorothy J. Glancy. *The Invention of the Right to Privacy*. Archived 2010-07-22 at the Wayback Machine, Arizona Law Review, 1979.

[131] Jeremy Goecks, W. Keith Edwards, and Elizabeth D. Mynatt. "Challenges in Supporting End-user Privacy and Security Management with Social Navigation". In: *Proceedings of the 5th Symposium on Usable Privacy and Security*. SOUPS '09. ACM, 2009, 5:1–5:12.

[132] Jennifer Golbeck et al. "Predicting Personality from Twitter." In: *SocialCom/-PASSAT*. IEEE, 2011, pp. 149–156.

[133] Lewis R. Goldberg. "Language and individual differences: The search for universals in personality lexicons". In: *Reviews of Personality and Social Psychology* (1981). SAGE Journals, pp. 159–181.

[134] Jorge Goncalves, Vassilis Kostakos, and Jayant Venkatanathan. "Narrowcasting in social media: Effects and perceptions". In: *IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*. IEEE, 2013, pp. 502–509.

[135] Felipe González et al. "Global Reactions to the Cambridge Analytica Scandal: A Cross-Language Social Media Study". In: *Companion Proceedings of The World Wide Web Conference*. WWW '19. ACM, 2019, pp. 799–806.

[136] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. Adaptive Computation and Machine Learning series. MIT Press, 2016.

[137] Samuel D. Gosling, Peter J. Rentfrow, and William B. Swann. "A Very Brief Measure of the Big-Five Personality Domains". In: *Journal of Research in Personality* 37.6 (2003). Elsevier, pp. 504–528.

[138] Mary J. Granger and Joyce Currie Little. "Classroom Discussions: Policies and Responsibilities of Internet Service Providers". In: *Proceedings of the 8th Annual Conference on Innovation and Technology in Computer Science Education*. ITiCSE '03. ACM, 2003, pp. 99–103.

[139] Ralph Gross and Alessandro Acquisti. "Information Revelation and Privacy in Online Social Networks". In: *Proceedings of the ACM Workshop on Privacy in the Electronic Society*. WPES '05. ACM, 2005, pp. 71–80.

[140] Jens Grossklags and Nigel J. Barradale. "Social Status and the Demand for Security and Privacy". In: *Privacy Enhancing Technologies*. Springer, 2014, pp. 83–101.

[141] Shumin Guo and Keke Chen. "Mining Privacy Settings to Find Optimal Privacy-Utility Tradeoffs for Social Network Services". In: *Proceedings of the ASE/IEEE International Conference on Social Computing and ASE/IEEE International Conference on Privacy, Security, Risk and Trust*. SOCIALCOM-PASSAT '12. IEEE, 2012, pp. 656–665.

[142] Sumana Gupta. "Design and Delivery of Medical Devices for Home-use: Drivers and Challenges". In: *3rd Institution of Engineering and technology International Conference on Medical Electrical Devices and Technology*. MEDTECH. 2007, pp. 215–235.

[143] Surbhi Gupta, Abhishek Singhal, and Akanksha Kapoor. "A literature survey on social engineering attacks: Phishing attack". In: *International Conference on Computing, Communication and Automation (ICCCA)*. IEEE, 2016, pp. 537–540.

[144] Tzipora Halevi, James Lewis, and Nasir Memon. "A Pilot Study of Cyber Security and Privacy Related Behavior and Personality Traits". In: *Proceedings of the 22Nd International Conference on World Wide Web*. WWW '13 Companion. ACM, 2013, pp. 737–744.

[145] Tzipora Halevi et al. "Cultural and Psychological Factors in Cyber-security". In: *Proceedings of the 18th International Conference on Information Integration and Web-based Applications and Services*. iiWAS '16. ACM, 2016, pp. 318–324.

[146] Keith Hampton and Barry Wellman. "Neighboring in Netville: How the Internet Supports Community and Social Capital in a Wired Suburb". In: *City & Community* 2.4 (2003). Wiley, pp. 277–311.

[147] Marian Harbach et al. "Using Personal Examples to Improve Risk Communication for Security & Privacy Decisions". In: *Proceedings of the 32nhistoryd Annual ACM Conference on Human Factors in Computing Systems*. CHI '14. ACM, 2014, pp. 2647–2656.

[148] Sarah Harris and David Harris. *Digital Design and Computer Architecture: ARM Edition*. 1st ed. Morgan Kaufmann Publishers, 2015.

[149] Mike Hart. "Do Online Buying Behaviour and Attitudes to Web Personalization Vary by Age Group?" In: *Proceedings of the 2008 Annual Research Conference of the South African Institute of Computer Scientists and Information Technologists on IT Research in Developing Countries: Riding the Wave of Technology*. SAICSIT '08. ACM, 2008, pp. 86–93.

[150] Marc Hassenzahl, Michael Burmester, and Franz Koller. "AttrakDiff: Ein Fragebogen zur Messung wahrgenommener hedonischer und pragmatischer Qualitaet". In: *Mensch & Computer 2003: Interaktion in Bewegung*. B. G. Teubner, 2003, pp. 187–196.

[151] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The elements of statistical learning: data mining, inference and prediction*. 2nd ed. Springer, 2009.

[152] D. Heckmann. *Ubiquitous User Modeling*. Dissertationen zur künstlichen Intelligenz - DISKI. Akademische Verlagsgesellschaft, 2006.

[153] Karen Henricksen, Ryan Wishart, and Ted McFadden. "Extending context models for privacy in pervasive computing environments". In: *2nd International Workshop on Contex Modelling and Reasoning (COMOREA), PERCOM'05 Workshop proceedings*. IEEE, 2005, pp. 20–24.

[154] Jonathan L. Herlocker, Joseph A. Konstan, and John Riedl. "Explaining Collaborative Filtering Recommendations". In: *Proceedings of the ACM Conference on Computer Supported Cooperative Work*. CSCW '00. ACM, 2000, pp. 241–250.

[155] Jan Hess et al. "In-situ Everywhere: A Qualitative Feedback Infrastructure for Cross Platform home-IT". In: *Proceedings of the 10th European Conference on Interactive TV and Video*. EuroITV '12. ACM, 2012, pp. 75–78.

[156] Richard F. Hixson. *Privacy in a Public Society: Human Rights in Conflict*. Oxford University Press, 1987.

[157] Christine Hogg and Charlotte Williamson. "Whose interests do lay people represent? Towards an understanding of the role of lay people as members of committees". In: *Health expectations : an international journal of public participation in health care and health policy* 4 (2001). Wiley, pp. 2–9.

[158] Jatinder Hothi and Wendy Hall. "An Evaluation of Adapted Hypermedia Techniques Using Static User Modelling". In: *Proceedings of the Second Workshop on Adaptive Hypertext and Hypermedia*. ACM, 1998, pp. 45–50.

[159] Bryan Hubbell et al. "Understanding social and behavioral drivers and impacts of air quality sensor use". In: *The Science of the total environment* 621 (2017). Elsevier, pp. 886–894.

[160] Markus Huber et al. "Cheap and Automated Socio-technical Attacks Based on Social Networking Sites". In: *Proceedings of the 3rd ACM Workshop on Artificial Intelligence and Security*. AISec '10. ACM, 2010, pp. 61–64.

[161] Thomas Hupperich et al. "On the Robustness of Mobile Device Fingerprinting: Can Mobile Users Escape Modern Web-Tracking Mechanisms?" In: *Proceedings of the 31st Annual Computer Security Applications Conference*. ACSAC. ACM, 2015, pp. 191–200.

[162] Luke Hutton, Tristan Henderson, and Apu Kapadia. "Short Paper: "Here i Am, Now Pay Me!": Privacy Concerns in Incentivised Location-Sharing Systems". In: *Proceedings of the ACM Conference on Security and Privacy in Wireless & Mobile Networks*. WiSec '14. ACM, 2014, pp. 81–86.

[163] Giovanni Iachello et al. "Developing Privacy Guidelines for Social Location Disclosure Applications and Services". In: *Proceedings of the 2005 Symposium on Usable Privacy and Security*. SOUPS '05. ACM, 2005, pp. 65–76.

[164] "In the Face of Danger: Facial Recognition and the Limits of Privacy Law". In: *Harvard Law Review*. The Harvard Law Review Association, 2007, pp. 1870–1891.

[165] Qatrunnada Ismail et al. "Crowdsourced Exploration of Security Configurations". In: *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. CHI '15. ACM, 2015, pp. 467–476.

[166] Gareth James et al. *An Introduction to Statistical Learning: With Applications in R*. Springer, 2014.

[167] Yousra Javed and Mohamed Shehab. "How Do Facebookers Use Friendlists". In: *Proceedings of the International Conference on Advances in Social Networks Analysis and Mining*. ASONAM '12. IEEE, 2012, pp. 343–347.

[168] Lukasz Jedrzejczyk et al. "On the Impact of Real-time Feedback on Users' Behaviour in Mobile Location-sharing Applications". In: *Proceedings of the Sixth Symposium on Usable Privacy and Security*. SOUPS '10. ACM, 2010, 14:1–14:12.

[169] Lukasz Jedrzejczyk et al. "Privacy-shake: a haptic interface for managing privacy settings in mobile location sharing applications". In: *MobileHCI '10: Proceedings of the 12th International Conference on Human Computer Interaction with Mobile Devices and Services*. ACM, 2010, pp. 411–412.

[170] Jason Jerald. *The VR Book: Human-Centered Design for Virtual Reality*. ACM and Morgan & Claypool, 2016.

[171] Lei Jin et al. "Exploiting Users' Inconsistent Preferences in Online Social Networks to Discover Private Friendship Links". In: *Proceedings of the 13th Workshop on Privacy in the Electronic Society*. WPES '14. ACM, 2014, pp. 59–68.

[172] Oliver P. John and Sanjay Srivastava. "The Big Five Trait Taxonomy: History, Measurement, and Theoretical Perspectives". In: *Handbook of Personality: Theory and Research*. 2nd ed. Guilford Press, 1999, pp. 102–138.

[173] A. Johnson and N.A. Taatgen. "User modeling". In: *Handbook of human factors in web design*. Lawrence Erlbaum Associates, 2005, pp. 424–438.

[174] Simon Jones and Eamonn O'Neill. "Feasibility of Structural Network Clustering for Group-based Privacy Control in Social Networks". In: *Proceedings of the Sixth Symposium on Usable Privacy and Security*. SOUPS '10. ACM, 2010, 9:1–9:13.

[175] Yolanda Jordaan and Gene Van Heerden. "Online privacy-related predictors of Facebook usage intensity". In: *Computers in Human Behavior* 70 (2017). Elsevier, pp. 90–96.

[176] Patrick W. Jordan. *An Introduction To Usability*. CRC Press, 1998.

[177] Sanjay Kairam et al. "Talking in Circles: Selective Sharing in Google+". In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '12. ACM, 2012, pp. 1065–1074.

[178] Marius Kaminskas and Francesco Ricci. "Location-Adapted Music Recommendation Using Tags". In: *User Modeling, Adaption and Personalization*. Springer Berlin Heidelberg, 2011, pp. 183–194.

[179] Pamela Karr-Wisniewski, David Wilson, and Heather Richter-Lipford. "A new social order: Mechanisms for social network site boundary regulation". In: *Americas Conference on Information Systems, AMCIS*. 101. AIS, 2011.

[180] Michaela Kauer et al. "Improving Privacy Settings for Facebook by Using Interpersonal Distance As Criterion". In: *CHI Extended Abstracts on Human Factors in Computing Systems*. CHI EA '13. ACM, 2013, pp. 793–798.

[181] Patrick Gage Kelley, Lorrie Faith Cranor, and Norman Sadeh. "Privacy As Part of the App Decision-Making Process". In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '13. ACM, 2013, pp. 3393–3402.

[182] Patrick Gage Kelley et al. "A "Nutrition Label" for Privacy". In: *Proceedings of the 5th Symposium on Usable Privacy and Security*. SOUPS '09. ACM, 2009, 4:1–4:12.

[183] Patrick Gage Kelley et al. "A Conundrum of Permissions: Installing Applications on an Android Smartphone". In: *Proceedings of the 16th International Conference on Financial Cryptography and Data Security*. FC'12. Springer, 2012, pp. 68–79.

[184] Patrick Gage Kelley et al. "An Investigation into Facebook Friend Grouping". In: *Proceedings of the 13th IFIP TC 13 International Conference on Human-computer Interaction*. Vol. 3. INTERACT'11. Springer, 2011, pp. 216–233.

[185] Jennifer King. "Taken Out of Context: An Empirical Analysis of Westin's Privacy Scale". In: *10th Symposium On Usable Privacy and Security (SOUPS)*. USENIX Association, 2014, pp. 1–18.

[186] M Kiranmayi and N Maheswari. "Reducing Attribute Couplet Attack in Social Networks using Factor Analysis". In: *2018 International Conference on Recent Trends in Advance Computing (ICRTAC)*. 2018, pp. 212–217.

[187] Ron Kohavi. "A Study of Cross-validation and Bootstrap for Accuracy Estimation and Model Selection". In: *Proceedings of the 14th International Joint Conference on Artificial Intelligence*. Vol. 2. IJCAI'95. Morgan Kaufmann Publishers Inc., 1995, pp. 1137–1143.

[188] Ron Kohavi and George H. John. "Wrappers for feature subset selection". In: *Artificial Intelligence* 97.1-2 (1997). Elsevier, pp. 273–324.

[189] Ron Kohavi and Foster Provost. "Glossary of Terms". In: *Machine Learning* 30.2-3 (1998), pp. 271–274.

[190] Juliane Kokott and Christoph Sobotta. "The distinction between privacy and data protection in the jurisprudence of the CJEU and the ECtHR". In: *International Data Privacy Law* 3.4 (2013). Oxford University Press, pp. 222–228.

[191] James Konow and Lars Schwettmann. "The Economics of Justice". In: *Handbook of Social Justice Theory and Research*. Springer New York, 2016, pp. 83–106.

[192] Michal Kosinski, David Stillwell, and Thore Graepel. "Private traits and attributes are predictable from digital records of human behavior". In: *Proceedings of the National Academy of Sciences of the United States of America* 110 (2013). National Academy of Sciences.

[193] John Krumm. "A survey of computational location privacy". In: *Personal and Ubiquitous Computing* 13.6 (2009). Springer, pp. 391–399.

[194] Max Kuhn and Kjell Johnson. *Applied predictive modeling*. Springer New York, 2013.

[195] Ponnurangam Kumaraguru and Lorrie Faith Cranor. *Privacy Indexes: A Survey of Westin's Studies*. Tech. rep. CMU-ISRI-5-138. Institute for Software Research International, School of Computer Science, Carnegie Mellon University, 2005.

[196] John Lamping, Ramana Rao, and Peter Pirolli. "A Focus+Context Technique Based on Hyperbolic Geometry for Visualizing Large Hierarchies". In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '95. ACM Press/Addison-Wesley Publishing Co., 1995, pp. 401–408.

[197] Marc Langheinrich. "A Privacy Awareness System for Ubiquitous Computing Environments". In: *Proceedings of the 4th International Conference on Ubiquitous Computing*. UbiComp '02. Springer, 2002, pp. 237–245.

[198] Yann LeCun et al. "Backpropagation Applied to Handwritten Zip Code Recognition". In: *Neural Computing* 1.4 (1989). MIT Press, pp. 541–551.

[199] Scott Lederer, Jennifer Mankoff, and Anind K. Dey. "Who Wants to Know What when? Privacy Preference Determinants in Ubiquitous Computing". In: *CHI Extended Abstracts on Human Factors in Computing Systems*. CHI EA '03. ACM, 2003, pp. 724–725.

[200] Scott Lederer et al. "Personal privacy through understanding and action: five pitfalls for designers". In: *Personal and Ubiquitous Computing* 8.6 (2004). Springer, pp. 440–454.

[201] Kun Chang Lee and Namho Chung. "Empirical analysis of consumer reaction to the virtual reality shopping mall". In: *Computers in Human Behavior* 24.1 (2008). Elsevier, pp. 88–104.

[202] Ming-Chi Lee. "Factors influencing the adoption of internet banking: An integration of TAM and TPB with perceived risk and perceived benefit". In: *Electronic Commerce Research and Applications* 8.3 (2009). Elsevier, pp. 130–141.

[203] Asko Lehmuskallio and Risto Sarvas. "Snapshot Video: Everyday Photographers Taking Short Video-Clips". In: *Proceedings of the 5th Nordic Conference on Human-Computer Interaction: Building Bridges*. NordiCHI '08. ACM, 2008, pp. 257–265.

[204] Bin Li, Qiang Yang, and Xiangyang Xue. "Can Movies and Books Collaborate?: Cross-domain Collaborative Filtering for Sparsity Reduction". In: *Proceedings of the 21st International Jont Conference on Artifical Intelligence*. IJCAI'09. Morgan Kaufmann Publishers Inc., 2009, pp. 2052–2057.

[205] Bin Li, Qiang Yang, and Xiangyang Xue. "Transfer Learning for Collaborative Filtering via a Rating-matrix Generative Model". In: *Proceedings of the 26th Annual International Conference on Machine Learning*. ICML '09. ACM, 2009, pp. 617–624.

[206] Jialiu Lin et al. "Modeling Users' Mobile App Privacy Preferences: Restoring Usability in a Sea of Permission Settings". In: *Symposium On Usable Privacy and Security (SOUPS)*. USENIX Association, 2014, pp. 199–212.

[207] Heather Richter Lipford, Andrew Besmer, and Jason Watson. "Understanding Privacy Settings in Facebook with an Audience View". In: *Proceedings of the 1st Conference on Usability, Psychology, and Security*. UPSEC'08. USENIX Association, 2008, 2:1–2:8.

[208] Bin Liu, Jialiu Lin, and Norman Sadeh. "Reconciling Mobile App Privacy and Usability on Smartphones: Could User Privacy Profiles Help?" In: *Proceedings of the 23rd International Conference on World Wide Web*. WWW '14. ACM, 2014, pp. 201–212.

[209] Bin Liu et al. "Follow My Recommendations: A Personalized Privacy Assistant for Mobile App Permissions". In: *Twelfth Symposium on Usable Privacy and Security (SOUPS)*. USENIX Association, 2016, pp. 27–41.

[210] Yabing Liu et al. "Simplifying Friendlist Management". In: *Proceedings of the Twenty-First International World Wide Web Conference (WWW'12)*. ACM, 2012.

[211] Stine Lomborg and Anja Bechmann. "Using APIs for Data Collection on Social Media". In: *The Information Society* 30.4 (2014). Routledge, pp. 256–265.

[212] Gustavo López, Gabriela Marin, and Marta Calderón. "Characterizing Ubiquitous Systems Privacy Issues by Gender and Age". In: *Proceedings of the 7th International Work-Conference on Ambient Assisted Living. ICT-based Solutions in Real Life Situations*. Vol. 9455. IWAAL. Springer, 2015, pp. 247–258.

[213] Yin Lu, Bernard Tan, and Kai-Lung Hui. "Inducing Customers to Disclose Personal Information to Internet Businesses with Social Adjustment Benefits". In: *ICIS Proceedings* 45 (2004). AIS.

[214] Er De Luca and Heinrich Hußmann. "Threat Awareness - Social Impacts of Privacy Aware Ubiquitous Computing". In: *European Cultural Studies Conference (INTER)*. Linköping University Electronic Press, 2007, pp. 1650–3686.

[215] Giuseppe Lugano and Pertti Saariluoma. "To Share or Not to Share: Supporting the User Decision in Mobile Social Software Applications". In: *User Modeling 2007*. Vol. 4511. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2007, pp. 440–444.

[216] Bengt-Åke Lundvall. *Product Innovation and User-Producer Interaction*. Aalborg Universitetsforlag, 1985.

[217] Heather MacNeil. *Without Consent: The Ethics of Disclosing Personal Information in Public Archives*. Society of American Archivists, 1992.

[218] Michelle Madejski, Maritza L. Johnson, and Steven M. Bellovin. "A study of privacy settings errors in an online social network." In: *PerCom Workshops*. IEEE, 2012, pp. 340–345.

[219] Gabriel Magno et al. "New Kid on the Block: Exploring the Google+ Social Graph". In: *Proceedings of the 2012 Internet Measurement Conference*. IMC '12. ACM, 2012, pp. 159–170.

[220] Michelle Majeski, Maritza Johnson, and Steven M. Bellovin. *The Failure of Online Social Network Privacy Settings*. Tech. rep. CUCS-010-11. Department of Computer Science, Columbia University, 2011.

[221] Naresh K. Malhotra, Sung S. Kim, and James Agarwal. "Internet Users' Information Privacy Concerns (IUIPC): The Construct, the Scale, and a Causal Model". In: *Information Systems Research* 15.4 (2004). INFORMS, pp. 336–355.

[222] David Maltz and Kate Ehrlich. "Pointing the Way: Active Collaborative Filtering". In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '95. ACM Press/Addison-Wesley Publishing Co., 1995, pp. 202–209.

[223] Charles H. Marler. "Privacy in a Public Society: Human Rights in Conflict". In: *American Journalism* 5.1 (1988). Routledge, pp. 55–56.

[224] Alessandra Mazzia, Kristen LeFevre, and Eytan Adar. "The PViz Comprehension Tool for Social Network Privacy Settings". In: *Proceedings of the Eighth Symposium on Usable Privacy and Security*. SOUPS '12. ACM, 2012, 13:1–13:12.

[225] Mindi McDowell and Damon Morda. "Socializing securely: using social networking services". In: *United States Computer Emergency Readiness Team* (2011). US-CERT.

[226] Jacqueline J. Meulman. "Prediction and classification in nonlinear data analysis: Something old, something new, something borrowed, something blue". In: *Psychometrika* 68.4 (2003). Springer, pp. 493–517.

[227] Martijn Millecamp et al. "Controlling Spotify Recommendations: Effects of Personal Characteristics on Music Recommender User Interfaces". In: *Proceedings of the 26th Conference on User Modeling, Adaptation and Personalization*. UMAP '18. ACM, 2018, pp. 101–109.

[228] Max Mills. "Sharing privately: the effect publication on social media has on expectations of privacy". In: *Journal of Media Law* 9.1 (2017). Routledge, pp. 45–71.

[229] Tehila Minkus and Keith W. Ross. "I Know What You're Buying: Privacy Breaches on eBay". In: *Privacy Enhancing Technologies*. Springer, 2014, pp. 164–183.

[230] Mainack Mondal et al. "Understanding and Specifying Social Access Control Lists". In: *Symposium On Usable Privacy and Security (SOUPS)*. USENIX Association, 2014, pp. 271–283.

[231] James H. Moor. "Towards a Theory of Privacy in the Information Age". In: *SIGCAS Computers and Society* 27.3 (1997). ACM, pp. 27–32.

[232] Barrington Moore. *Privacy: studies in social and cultural history*. M.E. Sharpe, 1984.

[233] Anthony Morton and M. Angela Sasse. "Privacy is a Process, Not a PET: A Theory for Effective Privacy Practice". In: *Proceedings of the 2012 New Security Paradigms Workshop*. NSPW '12. ACM, 2012, pp. 87–104.

[234] Allen Newell. *Unified Theories of Cognition*. Harvard University Press, 1990.

[235] Mark Newman and Michelle Girvan. "Finding and Evaluating Community Structure in Networks". In: *Physical Review E, Statistical, nonlinear, and soft matter physics* 69 (2004). American Physical Society, p. 026113.

[236] Norman H. Nie. "Sociability, Interpersonal Relations, and the Internet: Reconciling Conflicting Findings". In: *American Behavioral Scientist* 45.3 (2001). SAGE Journals, pp. 420–435.

[237] Helen Nissenbaum. *Privacy in Context: Technology, Policy, and the Integrity of Social Life*. Stanford University Press, 2019.

[238] Dara O'Neil. "Analysis of Internet Users' Level of Online Privacy Concerns". In: *Social Science Computer Review* 19 (2001). SAGE Journals, pp. 17–31.

[239] Kenneth Olmstead and Michelle Atkinson. *An Analysis of Android App Permissions.* Pew Research Center, 2015.

[240] Kenneth Olmstead and Michelle Atkinson. *The Next Web. Android Users Have an Average of 95 Apps Installed on Their Phones, According to Yahoo Aviate Data.* Pew Research center, 2015.

[241] Stanisław Osiński and Dawid Weiss. "Carrot2: An Open Source Framework for Search Results Clustering". In: *Proceedings of the 26th European Conference on Information Retrieval*. Springer, 2004, pp. 13–14.

[242] Daniel J. Ozer and Verónica Benet-Martínez. "Personality and the Prediction of Consequential Outcomes". In: *Annual Review of Psychology* 57.1 (2006). Annual Reviews, pp. 401–421.

[243] Weike Pan et al. "Transfer Learning to Predict Missing Ratings via Heterogeneous User Feedbacks". In: *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence*. Vol. 3. IJCAI'11. AAAI Press, 2011, pp. 2318–2323.

[244] Malcolm R. Parks and Kory Floyd. "Making Friends in Cyberspace". In: *Journal of Computer-Mediated Communication* 1.4 (1996). Oxford University Press.

[245] Sameer Patil, Xinru Page, and Alfred Kobsa. "With a Little Help from My Friends: Can Social Navigation Inform Interpersonal Privacy Preferences?" In: *Proceedings of the ACM Conference on Computer Supported Cooperative Work*. CSCW '11. ACM, 2011, pp. 391–394.

[246] Sameer Patil et al. "My Privacy Policy: Exploring End-user Specification of Free-form Location Access Rules". In: *Financial Cryptography and Data Security: FC 2012 Workshops, USEC and WECSR 2012*. Springer Berlin Heidelberg, 2012, pp. 86–97.

[247] Thomas Paul, Daniel Puscher, and Thorsten Strufe. "Improving the Usability of Privacy Settings in Facebook". In: *Computing Research Repository (CoRR)* abs/1109.6046 (2011). arXiv.

[248] Alexis M. Peddy. "Dangerous Classroom "App"-titude: Protecting Student Privacy from Third-Party Educational Service Providers". In: *Brigham Young University Education and Law Journal*. Vol. 2017. 5. bepress, 2017.

[249] James W Pennebaker, Martha E Francis, and Roger J Booth. *Linguistic inquiry and word count: LIWC 2001*. Mahway: Lawrence Erlbaum Associates, 2001.

[250] Marjani Peterson and Chutima Boonthum-Denecke. "Investigation of Cookie Vulnerabilities: Poster". In: *Proceedings of the 12th Conference on Security and Privacy in Wireless and Mobile Networks*. WiSec '19. ACM, 2019, pp. 330–331.

[251] Joseph Phelps, Glen Nowak, and Elizabeth Ferrell. "Privacy Concerns and Consumer Willingness to Provide Personal Information". In: *Journal of Public Policy & Marketing* 19.1 (2000). Cambridge University Press, pp. 27–41.

[252] Jo Pierson and Rob Heyman. "Social media and cookies: Challenges for online privacy". In: *The Journal of Policy, Regulation and Strategy for Telecommunications, Information and Media.* 13 (2011). Emerald Group Publishing, pp. 30–42.

[253] W. Plant and G. Schaefer. "Navigation and Browsing of Image Databases". In: *International Conference of Soft Computing and Pattern Recognition*. Springer, 2009, pp. 750–755.

[254] William Plant and Gerald Schaefer. "Visualisation and Browsing of Image Databases". In: *Multimedia Analysis, Processing and Communications*. Springer Berlin Heidelberg, 2011, pp. 3–57.

[255] David Pogue. "TechnoFiles: Don't Worry about Who's Watching". In: *Scientific American* 304.1 (2011). Springer, pp. 32–32.

[256] Sören Preibusch. "Guide to measuring privacy concern: Review of survey and observational instruments". In: *International Journal of Human-Computer Studies* 71.12 (2013). Elsevier, pp. 1133–1143.

[257] Aaron Quigley. "Large Scale 3D Clustering and Abstraction". In: *Selected Papers from the Pan-Sydney Workshop on Visualisation*. Vol. 2. VIP '00. Australian Computer Society, 2001, pp. 117–118.

[258] Aaron Quigley and Peter Eades. "FADE: Graph Drawing, Clustering, and Visual Abstraction". In: *Graph Drawing*. Springer Berlin Heidelberg, 2001, pp. 197–210.

[259] Kelly Quinn. "Why We Share: A Uses and Gratifications Approach to Privacy Regulation in Social Media Use". In: *Journal of Broadcasting & Electronic Media* 60.1 (2016). Routledge, pp. 61–86.

[260] F. Raber and A. Krüger. "Deriving Privacy Settings for Location Sharing: Are Context Factors Always the Best Choice?" In: *2018 IEEE Symposium on Privacy-Aware Computing (PAC)*. IEEE, 2018, pp. 86–94.

[261] Frederic Raber and Antonio Krüger. "Applications for In-Situ Feedback on Social Network Notifications". In: *17th IFIP TC 13 International Conference on Human-Computer Interaction*. INTERACT. Springer, 2019, pp. 654–658.

[262] Frederic Raber and Antonio Krüger. "OmniWedges: Improved Radar-Based Audience Selection for Social Networks". In: *17th IFIP TC 13 International Conference on Human-Computer Interaction*. INTERACT. Springer, 2019, pp. 654–658.

[263] Frederic Raber and Antonio Krüger. "Privacy Perceiver: Using Social Network Posts to Derive Users' Privacy Measures". In: *Adjunct Publication of the 26th Conference on User Modeling, Adaptation and Personalization*. UMAP '18. ACM, 2018, pp. 227–232.

[264] Frederic Raber and Antonio Krüger. "Towards Understanding the Infuence of Personality on Mobile App Permission Settings". In: *16th IFIP TC 13 International Conference on Human-Computer Interaction*. INTERACT. Springer, 2017.

[265] Frederic Raber and Antonio Krüger. "Transferring Recommendations through Privacy User Models across Domains". In: *User Modeling and User-Adapted Interaction* (). Springer.

[266] Frederic Raber, Christopher Schommer, and Antonio Krüger. "FriendGroupVR: Design Concepts Using Virtual Reality to Organize Social Network Friends". In: *17th IFIP TC 13 International Conference on Human-Computer Interaction*. INTERACT. Springer, 2019, pp. 654–658.

[267] Frederic Raber and Nils Vossebein. "URetail: Privacy User Interfaces for Intelligent Retail Stores". In: *16th IFIP TC 13 International Conference on Human-Computer Interaction*. INTERACT. Springer, 2017, pp. 473–477.

[268] Frederic Raber, David Ziemann, and Antonio Krüger. "The 'Retailio' Privacy Wizard: Assisting Users with Privacy Settings for Intelligent Retail Stores". In: *3rd European Workshop on Usable Security*. EuroUSEC. Internet Society, 2018.

[269] Frederic Raber et al. "Fine-grained Privacy Setting Prediction using a Privacy Attitude Questionnaire and Machine Learning". In: *16th IFIP TC 13 International Conference on Human-Computer Interaction*. INTERACT. IFIP. Springer, 2017.

[270] Emilee Rader and Rebecca Gray. "Understanding User Beliefs About Algorithmic Curation in the Facebook News Feed". In: *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. CHI '15. ACM, 2015, pp. 173–182.

[271] Hootan Rashtian et al. "To Befriend Or Not? A Model of Friend Request Acceptance on Facebook". In: *10th Symposium On Usable Privacy and Security (SOUPS)*. USENIX Association, 2014, pp. 285–300.

[272] Ramprasad Ravichandran et al. "Capturing Social Networking Privacy Preferences: Can Default Policies Help Alleviate Tradeoffs Between Expressiveness and User Burden?" In: *Proceedings of the 5th Symposium on Usable Privacy and Security*. SOUPS '09. ACM, 2009, 47:1–47:1.

[273] Robert W. Reeder et al. "A User Study of the Expandable Grid Applied to P3P Privacy Policy Visualization". In: *Proceedings of the 7th ACM Workshop on Privacy in the Electronic Society*. WPES '08. ACM, 2008, pp. 45–54.

[274] Payam Refaeilzadeh, Lei Tang, and Huan Liu. "Cross-Validation". In: *Encyclopedia of Database Systems*. Springer US, 2009, pp. 532–538.

[275] Jeffrey H. Reiman. "Privacy, Intimacy, and Personhood". In: *Philosophy and Public Affairs* 6.1 (1976). Wiley-Blackwell, pp. 26–44.

[276] Delphine Reinhardt, Franziska Engelmann, and Matthias Hollick. "Can I Help You Setting Your Privacy? A Survey-based Exploration of Users' Attitudes towards Privacy Suggestions". In: *Proceedings of the 13th ACM International Conference on Advances in Mobile Computing and Multimedia (MoMM)*. ACM, 2015.

[277] Bernardo Reynolds et al. "Sharing Ephemeral Information in Online Social Networks: Privacy Perceptions and Behaviours". In: *Proceedings of the 13th IFIP TC 13 International Conference on Human-computer Interaction*. Vol. 3. INTERACT'11. Springer, 2011, pp. 204–215.

[278] Alexandra Roshchina, John Cardiff, and Paolo Rosso. "A Comparative Evaluation of Personality Estimation Algorithms for the Twin Recommender System". In: *Proceedings of the 3rd International Workshop on Search and Mining User-generated Contents*. SMUC '11. ACM, 2011, pp. 11–18.

[279] Luca Rossi and Mirco Musolesi. "It's the Way You Check-in: Identifying Users in Location-Based Social Networks". In: *Proceedings of the Second ACM Conference on Online Social Networks*. COSN '14. ACM, 2014, pp. 215–226.

[280] Gordon Rugg and Peter McGeorge. "The sorting techniques: a tutorial paper on card sorts, picture sorts and item sorts". In: *Expert Systems* 14.2 (1997). Wiley, pp. 80–93.

[281] Norman Sadeh et al. "Understanding and Capturing People's Privacy Policies in a Mobile Social Networking Application". In: *Personal Ubiquitous Computing* 13.6 (2009). Springer, pp. 401–412.

[282] Shaghayegh Sahebi and Peter Brusilovsky. "Cross-Domain Collaborative Recommendation in a Cold-Start Context: The Impact of User Profile Size on the Quality of Recommendation". In: *User Modeling, Adaptation, and Personalization*. Springer Berlin Heidelberg, 2013, pp. 289–295.

[283] Mostafa Salama et al. "Computational Social Networks: Security and Privacy". In: *Computational Social Networks: Security and Privacy*. Springer London, 2012, pp. 3–21.

[284] Florian Schaub et al. "Privacy Context Model for Dynamic Privacy Adaptation in Ubiquitous Computing". In: *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*. UbiComp '12. ACM, 2012, pp. 752–757.

[285] Andrew I. Schein et al. "Methods and Metrics for Cold-start Recommendations". In: *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '02. ACM, 2002, pp. 253–260.

[286] Roman Schlegel, Apu Kapadia, and Adam J. Lee. "Eyeing Your Exposure: Quantifying and Controlling Information Sharing for Improved Privacy". In: *Proceedings of the Seventh Symposium on Usable Privacy and Security*. SOUPS '11. ACM, 2011, 14:1–14:14.

[287] Michiel Schwarz and Michael Thompson. *Divided We Stand: Redefining Politics, Technology, and Social Choice*. University of Pennsylvania Press, 1990.

[288] Marcello Paolo Scipioni and Marc Langheinrich. "Towards a new privacy-aware location sharing platform". In: *Journal of Internet Services and Information Security* (). Innovative Information Science & Technology Research Group, p. 2011.

[289] Gini G. Scott. *The Death of Privacy: The Battle for Personal Privacy in the Courts, the Media, and Society*. iUniverse, 2008.

[290] Norbert Seyff, Gregor Ollmann, and Manfred Bortenschlager. "AppEcho: A User-driven, in Situ Feedback Approach for Mobile Platforms and Applications". In: *Proceedings of the 1st International Conference on Mobile Software Engineering and Systems*. MOBILESoft. ACM, 2014, pp. 99–108.

[291] Guy Shani, Max Chickering, and Christopher Meek. "Mining Recommendations from the Web". In: *Proceedings of the ACM Conference on Recommender Systems*. RecSys '08. ACM, 2008, pp. 35–42.

[292] Kim Bartel Sheehan. "An investigation of gender differences in on-line privacy concerns and resultant behaviors". In: *Journal of Interactive Marketing* 13.4 (1999). Elsevier, pp. 24–38.

[293] Kim Bartel Sheehan. "Toward a Typology of Internet Users and Online Privacy Concerns". In: *The Information Society* 18.1 (2002). Routledge, pp. 21–32.

[294] Kim Bartel Sheehan and Mariea Grubbs Hoy. "Dimensions of Privacy Concern among Online Consumers". In: *Journal of Public Policy & Marketing* 19.1 (2000). Cambridge University Press, pp. 62–73.

[295] Mohamed Shehab and Hakim Touati. "Semi-Supervised Policy Recommendation for Online Social Networks". In: *Proceedings of the International Conference on Advances in Social Networks Analysis and Mining*. ASONAM '12. IEEE, 2012, pp. 360–367.

[296] Mohamed Shehab et al. "User Centric Policy Management in Online Social Networks". In: *International Symposium on Policies for Distributed Systems and Networks*. POLICY. IEEE, 2010, pp. 9–13.

[297] Nathan W. Shock, Gerontology Research Center (U.S.), and National Institute on Aging. *Normal Human Aging: The Baltimore Longitudinal Study of Aging*. DHHS publication. U.S. Department of Health and Human Services, Gerontology Research Center, 1984.

[298] Reza Shokri et al. "Protecting Location Privacy: Optimal Strategy Against Localization Attacks". In: *Proceedings of the ACM Conference on Computer and Communications Security*. CCS '12. ACM, 2012, pp. 617–627.

[299] Teerapol Silawan and Chaodit Aswakul. "SybilVote: Formulas to Quantify the Success Probability of Sybil Attack in Online Social Network Voting". In: *IEEE Communications Letters* 21.7 (2017). IEEE, pp. 1553–1556.

[300] Arunesh Sinha, Yan Li, and Lujo Bauer. "What You Want is Not What You Get: Predicting Sharing Policies for Text-based Content on Facebook". In: *Proceedings of the ACM Workshop on Artificial Intelligence and Security*. AISec '13. ACM, 2013, pp. 13–24.

[301] Rashmi Sinha and Kirsten Swearingen. "The Role of Transparency in Recommender Systems". In: *CHI Extended Abstracts on Human Factors in Computing Systems*. CHI EA '02. ACM, 2002, pp. 830–831.

[302] Mel Slater. "Place Illusion and Plausibility Can Lead to Realistic Behaviour in Immersive Virtual Environments". In: *Philosophical transactions of the Royal Society of London. Series B, Biological sciences* 364 (2009). British Royal Society, pp. 3549–57.

[303] H. Jeff Smith and Sandra J. Milberg. "Information Privacy: Measuring Individuals' Concerns About Organizational Practices". In: *MIS Quarterly* 20.2 (1996). Society for Information Management and The Management Information Systems Research Center, pp. 167–196.

[304] Dagobert Soergel, Tony Tse, and Laura Slaughter. "Helping Healthcare Consumers Understand: An "Interpretive Layer" for Finding and Making Sense of Medical Information". In: *Studies in health technology and informatics* 107 (2004). IOS Press, pp. 931–5.

[305] Daniel J. Solove. *Understanding Privacy*. 1st. Harvard University Press, 2008.

[306] Daniel J. Solove, Marc Rotenberg, and Paul M. Schwartz. *Privacy, Information, and Technology*. Aspen Elective. Aspen Publishers, 2006.

[307] Ashkan Soltani et al. "Flash Cookies and Privacy". In: *SSRN Electronic Journal* (2009). Elsevier.

[308] Lübomira Spassova et al. "Innovative Retail Laboratory". In: *Roots for the Future of Ambient Intelligence. European Conference on Ambient Intelligence (AmI-09)*. Springer, 2009.

[309] Marco Speicher et al. "A Virtual Reality Shopping Experience Using the Apartment Metaphor". In: *Proceedings of the 2018 International Conference on Advanced Visual Interfaces*. AVI '18. ACM, 2018, 17:1–17:9.

[310] Sarah Spiekermann, Jens Grossklags, and Bettina Berendt. "E-privacy in 2Nd Generation E-commerce: Privacy Preferences Versus Actual Behavior". In: *Proceedings of the 3rd ACM Conference on Electronic Commerce*. EC '01. ACM, 2001, pp. 38–47.

[311] Tasos Spiliotopoulos, Diogo Pereira, and Ian Oakley. "Predicting Tie Strength with the Facebook API". In: *Proceedings of the 18th Panhellenic Conference on Informatics*. PCI '14. ACM, 2014, 9:1–9:5.

[312] Anna C. Squicciarini, Mohamed Shehab, and Joshua Wede. "Privacy policies for shared content in social network sites". In: *The VLDB Journal* 19.6 (2010). Springer, pp. 777–796.

[313] James W. Stamos and David K. Gifford. "Remote Evaluation". In: *Transactions on Programming Languages and Systems* 12.4 (1990). ACM, pp. 537–564.

[314] Kathy A. Stewart and Albert H. Segars. "An Empirical Examination of the Concern for Information Privacy Instrument". In: *Information Systems Research* 13.1 (2002). INFORMS, pp. 36–49.

[315] Anselm Strauss and Juliet M. Corbin. *Basics of Qualitative Research : Techniques and Procedures for Developing Grounded Theory*. SAGE Publications, 1998.

[316] Fred Stutzman, Ralph Gross, and Alessandro Acquisti. "Silent Listeners: The Evolution of Privacy and Disclosure on Facebook". In: *Journal of Privacy and Confidentiality 4 (2)* (2013). Labor Dynamics Institute.

[317] Martin Szomszor et al. "Semantic Modelling of User Interests Based on Cross-Folksonomy Analysis". In: *The Semantic Web - ISWC 2008*. Springer Berlin Heidelberg, 2008, pp. 632–648.

[318] Karen P. Tang et al. "Rethinking Location Sharing: Exploring the Implications of Social-driven vs. Purpose-driven Location Sharing". In: *Proceedings of the 12th ACM International Conference on Ubiquitous Computing*. UbiComp '10. ACM, 2010, pp. 85–94.

[319] "The problem of multicollinearity". In: *Understanding Regression Analysis*. Springer US, 1997, pp. 176–180.

[320] Andrey N. Tikhonov and Vasiliy Y. Arsenin. *Solutions of ill-posed problems*. V. H. Winston & Sons, 1977, pp. xiii+258.

[321] Eran Toch. "Super-Ego: A Framework for Privacy-sensitive Bounded Context-awareness". In: *Proceedings of the 5th ACM International Workshop on Context-Awareness for Self-Managing Systems*. CASEMANS '11. ACM, 2011, pp. 24–32.

[322] Eran Toch, Yang Wang, and Lorrie Faith Cranor. "Personalization and privacy: a survey of privacy risks and remedies in personalization-based systems". In: *User Modeling and User-Adapted Interaction* 22.1 (2012). Springer, pp. 203–220.

[323] Sabine Trepte and Philipp K. Masur. "Need for Privacy". In: *Encyclopedia of Personality and Individual Differences*. Springer, 2017, pp. 1–4.

[324] Ashish K. Tripathi et al. "Assessing Personality Using Demographic Information from Social Media Data". In: *Proceedings of the 2015 International Conference on Social Media & Society*. SMSociety '15. ACM, 2015, 10:1–10:7.

[325] Janice Y. Tsai et al. "Location-sharing technologies: Privacy risks and controls". In: *Conference on Communication, Information and Internet Policy (TPRC)*. SSRN, 2009.

[326] Janice Y. Tsai et al. "Who's Viewed You? The Impact of Feedback in a Mobile Location-Sharing Application". In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '09. ACM, 2009, pp. 2003–2012.

[327] Virpi Kristiina Tuunainen, Olli Pitkänen, and Marjaana Hovi. "Users' Awareness of Privacy on Online Social Networking Sites – Case Facebook". In: *BLED 2009 Proceedings* 42 (2009).

[328] Rajan Vaish et al. "Twitch Crowdsourcing: Crowd Contributions in Short Bursts of Time". In: *Proceedings of the 32Nd Annual ACM Conference on Human Factors in Computing Systems*. CHI '14. ACM, 2014, pp. 3645–3654.

[329] Narseo Vallina-Rodriguez et al. "Header Enrichment or ISP Enrichment? Emerging Privacy Threats in Mobile Networks". In: *Proceedings of the 2015 ACM SIGCOMM Workshop on Hot Topics in Middleboxes and Network Function Virtualization*. HotMiddlebox '15. ACM, 2015, pp. 25–30.

[330] Graham Vickery and Sacha Wunsch-Vincent. *Participative Web and User-Created Content: Web 2.0, Wikis and Social Networking*. 1st ed. OECD Publications. OECD publishing, 2007.

[331] Jonas Walter, Bettina Abendroth, and Nupur Agarwal. "PRICON: Self-determined Privacy in the Connected Car Motivated by the Privacy Calculus Model". In: *Proceedings of the 16th International Conference on Mobile and Ubiquitous Multimedia*. MUM '17. ACM, 2017, pp. 421–427.

[332] Yang Wang and Alfred Kobsa. "A PLA-based Privacy-enhancing User Modeling Framework and Its Evaluation". In: *User Modeling and User-Adapted Interaction* 23.1 (2013). Kluwer Academic Publishers, pp. 41–82.

[333] Yang Wang et al. "A Field Trial of Privacy Nudges for Facebook". In: *Proceedings of the 32nd Annual ACM Conference on Human Factors in Computing Systems*. CHI '14. ACM, 2014, pp. 2367–2376.

[334] Yang Wang et al. "Privacy Nudges for Social Media: An Exploratory Facebook Study". In: *Proceedings of the 22nd International Conference on World Wide Web*. WWW '13 Companion. ACM, 2013, pp. 763–770.

[335] Samuel D. Warren and Louis D. Brandeis. "The Right to Privacy". In: *Harvard Law Review* 4.5 (1890). Harvard University Press, pp. 193–220.

[336] Jason Watson, Andrew Besmer, and Heather Richter Lipford. "+Your Circles: Sharing Behavior on Google+". In: *Proceedings of the Eighth Symposium on Usable Privacy and Security*. SOUPS '12. ACM, 2012, 12:1–12:9.

[337] Barry Wellman et al. "Computer networks as social networks: Collaborative work, telework and virtual community". In: *Annual Review of Sociology* 22 (1996). Annual Reviews, pp. 213–238.

[338] Barry Wellman et al. "Does the Internet Increase, Decrease, or Supplement Social Capital?: Social Networks, Participation, and Community Commitment". In: *American Behavioral Scientist* 45.3 (2001). SAGE Journals, pp. 436–455.

[339] Sara J. Weston, Patrick L. Hill, and Joshua J. Jackson. "Personality Traits Predict the Onset of Disease". In: *Social Psychological and Personality Science* 6.3 (2015). SAGE Journals, pp. 309–317.

[340] Michael E. Wiklund and Stephen B. Wilcox. *Designing Usability into Medical Products*. Taylor & Francis, 2005.

[341] Dmitri Williams. "On and off the 'Net: Scales for Social Capital in an Online Era". In: *Journal of Computer-Mediated Communication* 11.2 (2017). Wiley, pp. 593–628.

[342] Raymond Williams. *Keywords: A Vocabulary of Culture and Society*. Oxford paperbacks. Oxford University Press, USA, 1985.

[343] Robin Williamson. "Forgive and Remember: Managing Medical Failure". In: *Journal of the Royal Society of Medicine* 97.3 (2004). SAGE Journals, pp. 147–148.

[344] Glenn D. Wilson. "Eysenck Personality Profiler". In: *Encyclopedia of Personality and Individual Differences*. Springer, 2016, pp. 1–3.

[345] Pinata Winoto and Tiffany Tang. "If You Like the Devil Wears Prada the Book, Will You also Enjoy the Devil Wears Prada the Movie? A Study of Cross-Domain Recommendations". In: *New Generation Computing* 26.3 (2008). Springer, pp. 209–225.

[346] Pamela Wisniewski, Heather Lipford, and David Wilson. "Fighting for my space: Coping mechanisms for SNS boundary regulation". In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM. 2012, pp. 609–618.

[347] Pamela Wisniewski et al. "Give Social Network Users the Privacy They Want". In: *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing*. CSCW '15. ACM, 2015, pp. 1427–1441.

[348] Chirayu Wongchokprasitti et al. "User Model in a Box: Cross-System User Model Transfer for Resolving Cold Start Problems". In: *User Modeling, Adaptation and Personalization*. Springer, 2015, pp. 289–301.

[349] Allison Woodruff et al. "Would a Privacy Fundamentalist Sell Their DNA for $1000... If Nothing Bad Happened as a Result? The Westin Categories, Behavioral Intentions, and Consequences". In: *Proceedings of the Tenth USENIX Conference on Usable Privacy and Security*. SOUPS '14. USENIX Association, 2014, pp. 1–18.

[350] Maomao Wu. "Adaptive privacy management for distributed applications". In: *Dissertation*. Lancaster University, 2007.

[351] Rongjing Xiang, Jennifer Neville, and Monica Rogati. "Modeling Relationship Strength in Online Social Networks". In: *Proceedings of the 19th International Conference on World Wide Web*. WWW '10. ACM, 2010, pp. 981–990.

[352] Heng Xu. "The Effects of Self-Construal and Perceived Control on Privacy Concerns." In: *Proceedings of 28th Annual International Conference on Information Systems (ICIS)*. AIS, 2007, p. 125.

[353] Runhua Xu et al. "Towards Understanding the Impact of Personality Traits on Mobile App Adoption - A Scalable Approach." In: *European Conference on Information Systems*. ECIS. AIS, 2015.

[354] Yuto Yamaguchi, Toshiyuki Amagasa, and Hiroyuki Kitagawa. "Landmark-based User Location Inference in Social Media". In: *Proceedings of the First ACM Conference on Online Social Networks*. COSN '13. ACM, 2013, pp. 223–234.

[355] Alyson L. Young and Anabel Quan-Haase. "Information Revelation and Internet Privacy Concerns on Social Network Sites: A Case Study of Facebook". In: *Proceedings of the Fourth International Conference on Communities and Technologies*. C&T '09. ACM, 2009, pp. 265–274.

[356] Qianyun Zhang, Shawndra Hill, and David Rothschild. "Post Purchase Search Engine Marketing". In: *Companion Proceedings of the The Web Conference*. WWW '18. ACM, 2018, pp. 663–670.

[357] Yu Zhang, Bin Cao, and Dit-Yan Yeung. "Multi-domain Collaborative Filtering". In: *Proceedings of the Twenty-Sixth Conference on Uncertainty in Artificial Intelligence*. UAI'10. AUAI Press, 2010, pp. 725–732.

[358] Marvin Zuckerman and C. Robert Cloninger. "Relationships between Cloninger's, Zuckerman's, and Eysenck's dimensions of personality". In: *Personality and Individual Differences* 21.2 (1996). Elsevier, pp. 283–285.

[359] Tomasz Zukowski and Irwin Brown. "Examining the Influence of Demographic Factors on Internet Users' Information Privacy Concerns". In: *Proceedings of the 2007 Annual Research Conference of the South African Institute of Computer Scientists and Information Technologists on IT Research in Developing Countries*. SAICSIT '07. ACM, 2007, pp. 197–204.