# Enriching input in Statistical Machine Translation

*Eleftherios Avramidis*


Master of Science
School of Informatics
University of Edinburgh
2007

# Abstract

Statistical Machine Translation has problems dealing with morphologically rich languages; translating from English into these gives a significantly worse quality. We make an effort to address this problem by adding per-word linguistic information into the source language of the translation task. We use the syntax of the source sentence so as to extract information for noun cases, verb persons and attribute genders and annotate these words accordingly. The solution is tested on factored phrase-based models, giving indications that the methods proposed are useful. Manual error analysis shows that the translation of the words annotated (nouns and verbs) improves, but a problem of sparse data is caused. Experiments managed to get a small improvement on NIST metric while human evaluation showed that a model combining both noun cases and verb persons has increased the adequacy (meaning) and deteriorated the fluency of the generated translation.

# Acknowledgements

Many thanks to my supervisor, Philipp Koehn, for the support and the experience he provided me with. Also thanks to the people of the Statistical Machine Translation Group of the Edinburgh University, for giving me the chance to grab some ideas from the papers they discussed. Special thanks to: Josh Schroeder who went through the whole trouble of preparing training and testing data, just to let us have the experiments run; Trevor Coehn for giving me a constructive feedback on the draft version of the write-up; Hieu Hoang for having all the answers on my questions for the decoding process; George Petasis who provided me with the essential annotated data for the Greek language.

Last but not least, many thanks to the thirteen kind Greek people who volunteered to participate in the human evaluation system without any reward.

# Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(*Eleftherios Avramidis*)

to my mother, who has provided me with motivation and funding all these years;
and to my brother with wishes for a good start in his studies

# Contents

.

# List of Figures

# List of Tables

# Chapter 1

# Introduction

## 1.1 Motivation

About 60 years after the first studies in automatic translation, much of the researchers' interest is nowadays focused on Statistical Machine Translation (SMT), which has employed widely used machine learning approaches in order to perform the translation task. During the latest years, SMT has evolved significantly, by incorporating a wide range of methods and improving the translation quality. Statistical language models have been effectively used in order to achieve good results and ongoing research is taking place so as to incorporate new capabilities and induce further enhancement.

As machine translation is applied on language pairs, crucial differences on the way the two languages operate can make the translation task too complicated. Due to its merely probabilistic basis being focused on a lexical level, SMT fails to produce adequate results in many cases which employ complicated linguistic phenomena. This stands mainly as a result of the incapability of the bare statistical systems to capture and model linguistic rules that cannot be directly "learnt" during the basic training.

Human languages worldwide are structured in different ways, concerning aspects such as syntax, grammar and the use of the vocabulary. Some languages are capable of encoding long meanings in simple ways, while others require more complicated structures. A very specific subset of the whole problem is the case of translating from a poor language, in terms of morphology, to a richer one. This mainly means that for a single word in the source language, there may be several translation candidates of the same target word, but appearing in different forms. Typically, the correct form of the inflected words should be chosen following several rules, e.g. depending on their syntactic role or their position in the sentence.

Figure 1.1 shows a motivating example, on how the same sentence would be perceived in each of our two languages. Greek requires many additional information, which are not given on a lexical level. All the information required for generating the output sentence are shown in brackets. Worst case, sequence-based models would fail to capture long dependencies and lead to misunderstandings. In our example, since there was no indication about any of the required information, the second Greek translation was incorrectly translated to say that "*articles got frustrated*".

**English**: **The president, after reading the press review and the articles, got frustrated**

**Greek**-1: **The president**[male,nominative], **after reading**[3rd sing] **the press review**[accusative,female,sing] **and the articles**[accusative,neutral,plural], **got**[3rd sing] **frustrated**[male,sing]

**Greek**-2: **The president**[male,nominative], **after reading**[3st sing] **the press review**[accusative,female,sing] **and the articles**[nominative,neutral,plur], **got**[3rd,plur] **frustrated**[neutral,plur]

Figure 1.1: Example of linguistic information needed in a language pair

## 1.2 Aims and scope

As the title indicates, this project is trying to deal with the described issue by "enriching" the translation input. Therefore, the main effort is to augment the words on the source side, by using linguistic information that may lead to better decisions to be taken while decoding. For this purpose, we will see how this process is based on the lately developed model of using factors during phrase-based statistical machine translation, and how an error analysis can reveal some of the aspects which we need to focus on.

Consequently, the hypothesis that is being examined is that raw source text does not always contain sufficient information for proceeding with the decoding, and therefore several ideas for enriching the input are part of the experiments. Our main aim is to indicate that using methods improves the appearance and the meaning of the translation outcome.

The main experimental basis is set on a one-way translation from English to Greek, based on the observation that both languages demonstrate linguistic behaviour on the

background that has been explained above. Our efforts will be based on factored phrase-based statistical machine translation models. Thus, we will focus on the part of preprocessing the source data so as to acquire the needed information and then use this data to train the models and compare their performance over a baseline system.

## 1.3 Chapters overview

As all of the experiments are based on phrase-based and factored translation models, the project begins on chapter 2, with introducing the basic background theory employed for designing the project. We give a detailed explanation of how the translation probabilities are estimated and how this is applied during the actual process. Finally we show the way the evaluation metrics are being calculated. Chapter 3 briefly reviews previous work on a similar issues, where research has taken place in order to tackle problems of unequal morphological richness in a translation pair. We briefly explain methods similar to ours and make some comparisons when this is feasible. Finally, we illustrate how our method tries to augment the previous contributions.

Chapter 4 gives the basic design for performing an experiment. Here we describe the framework of our system and we focus on the technical side of the set-up. As our framework is divided into steps, there is a brief explanation of each of them, so as to show how the model building and the evaluation process was performed. The actual methods that were used in order to add linguistic information are shown in Chapter 5. We focus on enriching input for three subsets of the problem: the nouns, the verbs and the attributes.

In Chapter 6 the experiments are presented in detail. All systems built are described and the outcomes of the various evaluation efforts are given. Manual error analysis and human evaluation are used in order to judge the effectiveness of the designed methods and lead to the conclusions which are summed up in Chapter 7, along with ideas for future work.

# Chapter 2

# Theoretical background

Theoretical research on Machine Translation (MT) is considered to have started during the late '40s, when Warren Weaver made the first efforts to "decode" foreign text (Trujillo, 1999), by applying statistical and cryptographic techniques developed for communication theory. Among the theoretical and practical efforts that followed these, we shall focus on the latest research on Statistical Machine Translation, which has had a rapid development during the last two decades. Following the path from phrase-based SMT (Marcu and Wong, 2002; Koehn et al., 2003) to factored translation models (Koehn and Hoang, 2007), this chapter presents the basic theory used for identifying the problems and designing the experiments shown in later chapters.

## 2.1 Phrased-based SMT

The first approach for Statistical Machine Translation (SMT) was given by the so-called IBM approach (Brown et al., 1990). This model, along with some refined IBM models published later (Brown et al., 1991, 1993; Brown, 1993) set the beginning of the modern research on SMT. The basic idea was that every source word was given a probability for being translated to every target word, while additional phenomena such as reordering were partially handled.

A clear disadvantage of the single-word based approach was the incapability of this statistical translation model to capture multi-word units, such as collocations, phrasal verbs etc. Therefore, the next step was the gradual application of the idea of a phrase-based model.

So, a significant improvement came when phrase based systems were implemented some years later. First, (Och et al., 1999) it became possible to model phrases so as

to conduct an alignment template model. Later on, (Marcu and Wong, 2002) the single IBM models were improved by calculating a joint-probability based on identifying phrases. Finally, (Koehn et al., 2003) a full decoding algorithm on phrase-based translation was introduced.

### 2.1.1 Training

The translation model is built based on a bilingual corpus, made by human translators. Researchers have found, so far, a good source of such resources in officially translated documents of multilingual state organisations, such as the proceedings of the Canadian Parliament (Brown et al., 1990) or the European Parliament (Koehn, 2005). The initial idea of this model also presumes that sentences in each language have been split and aligned in pairs where, in every pair, a source language sentence is aligned with its translation.

The phrase-based training process identifies all possible phrase pairs between a source sentence and a target sentence. Within each sentence, every source phrase is assigned a probability for being aligned to every phrase in the target sentence. A further repetition of this process along the training data essentially improves the probabilities based on the seen phrase translations.

Figure 2.1: Sample phrase based alignment for English to Greek, based on example of (Och et al., 1999)

At first, a basic phrase dictionary is constructed, in order to identify all possible phrase pairs. The phrase-based model uses the GIZA++ toolkit (Och and Ney, 2003) in order to perform the word-alignment task between the two sentences per pair, in both directions (Och et al., 1999). The union of these word-alignments (see figure 2.1) gives a symmetrised alignment matrix. Heuristics may be used in order to improve the alignment.

Secondly, the phrase translation probability is calculated by the observations on the training set. The basic relative frequency rule (for two languages **f** and **e**) would be:

$$\phi(\overline{e}|\overline{f}) = \frac{count(\overline{e}|\overline{f})}{\sum_{\overline{e}} count(\overline{e},\overline{f})} \tag{2.1}$$

The whole process leads to the creation of a phrase translation table, which describes how the seen phrase pairs are mapped.

### 2.1.2 Phrase-based model definition

Let's consider the case of translating from English to a foreign language, which will be presented as a common task later in this thesis. The basic theoretic approach uses the noisy channel approach, in order to maximise the translation probability from English **e** to a foreign language **f**.

$$\overline{f} = argmax\ p(f|e) = argmax\ p(e|f)p(f) \tag{2.2}$$

which is defined by the translation model $p(e|f)$ and a language model of the target language.

The foreign language f consists of $I$ phrases $\overline{f}_1^I$ and each of them is mapped from an English phrase $\overline{e}_i$ while a *distortion model* is used for achieving reordering in the target language. The best translation is therefore determined by:

$$f_{best} = argmax_f p(f|e) = argmax_f p(e|f)p_{LM}(f)\omega^{length(f)} \tag{2.3}$$

where

$$p(e|f) = p(\overline{e}_1^I|\overline{f}_1^I) = \prod_{i=1}^{I} \phi(\overline{e}_i|\overline{f}_i)d(a_i - b_{i-1}) \tag{2.4}$$

where $\phi(\overline{e}_i)$ models the translation probability distribution $d(a_i - b_{i-1})$ is the distortion model and $\omega$ is a factor empirically used to bias long output.

### 2.1.3 Decoding

The decoding process uses a beam search algorithm (Jelinek, 1997), which parses source input from left to right, and produces all possible partial translations of the phrases encountered in the input. The score of each hypothesis is calculated and the one with the highest score is chosen.

Figure 2.2: Efficient decoding using stacks (Koehn, 2007)

In order to effectively handle the translation options, as the hypotheses are being expanded, these are organised in *stacks* (figure 2.2), usually depending on the number of foreign words translated. During the decoding process, all hypotheses from one stack are expanded and are placed into further stacks. Since the search space may grow

significantly, given a long sentence, the algorithm performs pruning of the weakest hypotheses in each stack, keeping always the best *n* ones. For this purpose an estimation of the *future cost* (Koehn et al., 2003) is calculated, in order to predict the cost of translating the remaining part of input, given each partial translation. The translation model cost is looked up, while the language model cost is being estimated, since there is no prior context. The reordering model cost is ignored. As a result, if the cheapest expansion of the current partial translation is estimated to have too low a probability, then this hypothesis is not stored for further expansion.

## 2.2 Factored models

Phrase-based translation managed to overcome several problems that the word-based models faced on word ordering and common phrases. Though, in efforts to further improve accuracy, it became obvious that the single word surfaces, contained in the phrases, are not enough for capturing much of language behaviour.

The use of extra features per word, in order to enhance several Natural Language Processing tasks, has been a common practice lately. Since those tasks required information more than the words themselves could indicate, various machine learning models have been used in order to take additional annotation into consideration.

A similar application in SMT resulted into creating the "Factored model approach" (Koehn and Hoang, 2007) which allows the use of additional tags per words during the translation task. According to this approach, the notion of a word, as used in the phrase-based model, is now extended by using a vector of multiple factors instead of a single word. These factors are usually tags about additional properties of each word, which may indicate additional relations to be considered within a translation model. Ideally, a good selection of factors could result in modelling the linguistic rules that rely beyond the word surface of the given text.

As it will be detailed below, the use of factors deconstructs the execution of many SMT sub-tasks in multiple levels, depending on how they fit each problem better. The multi-level results are afterwards combined in order to generate a surface sentence as an outcome of these multiple underlying procedures.

### 2.2.1 The use of factors in training

Training in factored models is based on additional per-word annotation on the bilingual corpus. Most often, common tools or on-demand scripts are used to obtain the suitable tags (e.g. part-of-speech, word classes, morphological classes, syntax and various word features). In any case, tags are not required in all cases and can be omitted depending on the availability of linguistic resources.

The word alignment process is similar to the one used for the phrase-based models, as described above. In this case though, it is possible to perform the task on one (the surface of the word) or more factor levels.

Figure 2.3: Example factored model for morphological analysis and generation (Koehn and Hoang, 2007)

In order to perform the translation probability estimation, it is needed to define the way that the factor levels on the source side are mapped to the factor levels on the output side (fig. 2.3). Then, the phrase translation distribution for each of the factor mappings leads to a separate phrase translation table.

Finally, on the output, the *generation* step handles the way the various factor levels are combined into the basic word-surface. The generation distribution forms a separate generation table.

### 2.2.2 Factored model definition

The factored statistical machine translation model uses a log-linear model, in order to combine the several components, including the language model, the reordering model, the translation levels and the generation. The model is defined (Koehn and Hoang, 2007) as following:

$$p(\mathbf{f}|\mathbf{e}) = \frac{1}{Z}\exp\sum_{i=1}^{n}\lambda_i h_i(\mathbf{f},\mathbf{e}) \tag{2.5}$$

where $\lambda_i$ is a vector of weights determined during a tuning process, and $h_i$ is the feature function. The feature function is defined differently for each translation component. Therefore, for the translation probability distribution, we have:

$$h_T(\mathbf{f}|\mathbf{e}) = \sum_j \tau(\overline{e}_j, \overline{f}_j) \tag{2.6}$$

and for the generation step:

$$h_G(\mathbf{f}|\mathbf{e}) = \sum_k \gamma(e_k) \tag{2.7}$$

### 2.2.3 Decoding using factors

The decoding step is processed as in the phrase-based model, but essentially extended. The beam search algorithm is used in the same way, but now taking into consideration the translation options which are result of more than one phrase translation tables. In this case, there is a high possibility of being unable to handle the magnitude of the translation options exploration. Therefore, the decoding algorithm needs to be adapted as needed, so as to avoid such as issue, by employing efficient search methods (pruning/caching).

As an application of the factored decoding process, *Moses* (Koehn et al., 2007) has been developed; a tool which will be used as part of the experiments shown in this project.

Despite the large computational load that emerges from the big size of a typical translation table, Moses uses several techniques in order to improve performance in a more efficient way. Only the part of the translation table that is needed for a translation task, can be loaded, while there are several other characteristics such as a prefix-tree structure for source words and effective caching of the translation candidates. The exact way Moses was incorporated into the experimental framework will be examined later in the thesis (section 4.2, page 22).

## 2.3 Evaluating Statistical Machine Translation

Apart from the actual process of designing and implementing a refined SMT model, evaluating its performance is quite important. As it has been a common practice in scientific research for *task-based* evaluation efforts, the development of a model is based on a *baseline* system, which reflects a basic implementation without any alterations or parameters defined (e.g. with no factors used, in our case). Then, assumptions are specified by suggesting further modifications and parametrisation. The outcome of any new experimental system is compared to the one of the baseline and, if any significant improvement is shown, the assumption is considered to hold.

As it is obvious, that comparison requires a defined metric which would judge the translation quality and/or lead to a specific measure of the improvement. For the eval-

uation during this project, we concentrated on the state-of-the-art evaluation methods used in related experiments. Below, a brief theoretical background is given for each of them .

### 2.3.1 BLEU score

Given the difficulty in performing human evaluation, recent efforts have turned to the use of automatic machine translation methods. The *BLEU* scoring method (Papineni et al., 2001) has been one of the mostly used ones and is commonly well-appreciated since it correlates highly with human evaluation and has little marginal cost per run.

The main idea of BLEU uses a reference human translation of good quality, to be compared with the machine-generated translation of the same source. The scoring is based on the weighted average of variable length phrase matches between the SMT outcome and the reference. The evaluation algorithm counts the matches after comparing $n$-grams of the candidate with $n$-grams of the reference translation. In particular, as both texts are represented by $n$-grams, a series of candidate $n$-grams are produced. The evaluation system weights the maximum value of the candidates who are matching the reference $n$-grams by the total count of the candidate $n$-grams. The total *modified* precision score, $p_n$ is recalculated given the candidates on a whole block of text:

$$p_n = \frac{\sum_{C \in \{Candidates\}} \sum_{n-gram \in C} Count_{matched}(n-gram)}{\sum_{C\prime \in \{Candidates\}} \sum_{n-gram\prime \in C\prime} Count_{matched}(n-gram\prime)} \tag{2.8}$$

In order to better address the problems related to precision, by considering aspects such as the length of the candidate translation, the final BLEU score takes the geometric mean of the test corpus' modified precision scores, multiplied by a brevity penalty factor.

$$BP = \begin{cases} 1 & \text{if } c > r \\ e^{1-\frac{r}{c}} & \text{if } c <= r \end{cases} \tag{2.9}$$

then, using uniform weights $w_n$

$$BLEU = BPexp\left(\sum_{n=1}^{N} w_n log p_n\right) \tag{2.10}$$

### 2.3.2 NIST metric

The NIST metric (NIST, 2002) was defined one year after the BLEU metric, following the same idea. In contrast to the previous approach, though, the NIST score is cal-

culated by taking into account the information gain from each n-gram, instead of the n-gram precision. So, NIST gives a better score to an n-gram match that is difficult, than to the ones who are comparatively easier.

In detail, the information gain is calculated as:

$$Info(w_1...w_n) = log_2\left(\frac{number\ of\ occurrences\ of\ w_1...w_{n-1}}{number\ of\ occurrences\ of\ w_1...w_n}\right) \tag{2.11}$$

which modifies the BLEU formula as following:

$$Score = \sum_{n=1}^{N}\left\{\frac{\sum_{co-occ\ w_1...wn} Info(w_1...w_n)}{\sum_{all\ w_1...w_n}(1)}\right\}\cdot exp\left\{\beta log^2\left[min\left(\frac{L_{sys}}{\overline{L}_{ref}},1\right)\right]\right\} \tag{2.12}$$

In this formula, β is chosen as a modified brevity penalty, N is a constant, $\overline{L}_{ref}$ the number of words in a reference translation, averaged over all reference translations and $L_{sys}$ the number of words in the translation being scored .

### 2.3.3 Evaluation significance

There have been several efforts to explain and further judge the evaluation metrics described above. Analyses have shown that the are limitations and that not all of their behaviour is fully explicable (Zhang et al., 2004). One of the issues faced in this project had to do with the statistical significance of the scores achieved, in terms of comparing two systems.

For this reason, *pairewise bootstrap re-sampling* (Koehn, 2004) has been introduced in order to compute statistical significance of test results and validate whether the difference between the metric scores of two experiments give reason for assuming that one system is better than the other. The method performs a repetitive re-sampling procedure with replacement, in order to create a new set of sentences from the full set, and evaluation is performed on the new set, using both translation systems. If the metrics comparison of two systems, on the same set of sentences, holds for a percentage of the repetitions (equal to our confidence interval, e.g. 95%) then the comparison is considered to be statifically significant.

## 2.4 Conlusion

We have looked in the fundamental theory, on which development and experiments have been based. Phrase based models set a robust background for handling long

expressions and collocations, while factored models can consider more information than just that given by the word surface. In the followin chapter, we will proceed by explaining several approaches which are based on a similar background but specialise on taking advantage of syntax and morphology.

# Chapter 3

# Related work

This project uses several methods of employing syntax and morphology for enriching the SMT input, as it will be shown in the next chapters. Hereby, several pieces of work related to this subject are presented.

## 3.1 Syntax-based approaches

As syntax is definitely an important structural aspect of the languages that take part in the translation process, there have been some efforts to incorporate syntax into an SMT system. Since our effort includes using information related to syntax, it might be useful to see some SMT methods which used syntax as their basis.

One of the first well-known efforts to perform Syntax-based statistical machine translation (Yamada and Knight, 2001) uses syntax parsing in order to perform clause reordering. The translation operation takes place on each leaf of the syntax tree, assuming that it depends only on the word itself, and doesn't take any context into account. Therefore, this work and works that followed in this path (Collins et al., 2005; Huang et al., 2006; Wang et al., 2007) do not take advantage of syntax in terms of how it would affect the selection of the candidates on a word-level and therefore they don't deal with known morphology issues that are tightly coupled with syntax.

A later method (Koehn and Knight, 2003) achieved significant improvement by defining noun-phrases and prepositional phrases as subtask of the statistical machine translation and introducing syntactic features on them. The input side was described by its parse tree, derived during the detection of the subphrases. The candidate output was annotated with POS tags and the most likely syntax tree. The features used ranked positively the preservation of the number of nouns and their determiners along with

the correct transfer of the noun

A latest approach (Birch et al., 2007), includes an effort to use syntax hints in factored SMT models by attaching *Combinatorial Categorial Grammar* (CCG) *supertags* as factors on the input words. This approach seems closer to influencing the correct lexical choice based on the syntactic position of the source words. The experiments described were focused on translating into English and the results were not very conclusive.

## 3.2 Dealing with morphology

### 3.2.1 First efforts on morphology

Issues that arise when using SMT techniques for translating between languages which have unequal morphology, have been investigated years ago. In 1987, before even the statistical machine translation had been in application, (Nitta, 1986) identified the so-called "idiosyncratic gap", which is defined as the difference in the way two languages operate. They did so by applying a Cross Translation Test, a practical method which compared a literal (word to word) translation to a free human translation. This seems as one of the first efforts to identify and measure language differences in translated language pairs.

The creators of the first SMT model, did suggest (Brown et al., 1992) incorporating the results of a morphological analysis to their approach. They performed morphological analysis of the inflection in both languages (English and French) for verbs, nouns, adjectives and adverbs. The experiments on the single-word SMT were performed by replacing the inflected forms with stems and tags, and they were able to show improvement on a low-scale evaluation performed by humans.

Following research (Niessen and Ney, 2001, 2004; Lee, 2004; Goldwater and McClosky, 2005; El Isbihani et al., 2006) focused thoroughly on the morphology, mainly motivated by the problem of scarce resources. The experiments performed involved translating into English from French, German, Czech, Arabic etc., by examining the linguistic behaviour of these translation pairs.

### 3.2.2 Translating into morphologically rich languages

While there has been a lot of research for resolving the morphology disproportion when translating into English, there has been comparatively little effort for tackling the lack

of morphology when translating from English into morphologically richer languages. It has been apparent upon the construction of the Europarl corpus (Koehn, 2005) that translating into such richer languages is definitely more difficult than translating from them. Though, the international research community has been focused on translating into English and has consequently neglected interesting problems that arise on the inverse translation direction.

The issue of the verb persons (which in English are denoted by detached personal pronouns, while in other languages resort in the morphology of the verb word surface) coheres with a lot of approaches for Word Sense Disambiguation (WSD). One of them (Mitamura et al., 2002) used WSD techniques to resolve the person of the verb in an English-Spanish knowledge-based MT system. Despite knowledge-based MT systems stay far from the statistical framework of the current project, the process of resolving the verb persons seems useful an idea.

One of the first efforts to enrich input in SMT (Ueffing and Ney, 2003) used Part-of-Speech (POS) tags in order to deal with the verb person, in English-Spanish and English-Catalan translation tasks. The problem was figured on a word-to-word translation system, since that was unable to efficiently handle the Latin verb persons. POS tags were practically used to let the authors identify the pronoun+verb sequence and splice these two words into one term. Interestingly enough, a Maximum Entropy Model is used, so as to let the verb stem be translated indifferently of the pronoun, if as its spliced couple consist an unseen event. Though, this problem, as the one (Brown et al., 1992) described above, is clearly motivated by the problems occurring by a single-word-based SMT and have been solved by adopting a phrase-based model. Meanwhile, there is no handling of the case when the pronoun stays in distance with the related verb.

Minkov et al. (2007) suggested a model which uses morphological and syntactic features, in order to ensure grammatical agreement on the output. The features are employed by a probabilistic model and are tested both on a monolingual output level and on both sides. The source side features mark POS tags, person, number, gender, object and possessive pronouns, compounds and prepositions, highly adapted for the tasks of translating into Arabic and Russian. The method was not tested on an MT system, but directly on the reference translations, achieving higher accuracy.

Similarly, translating English into Turkish (Durgar El-Kahlout and Oflazer, 2006) has challenged the use of POS and morph stems in the input along with rich Turkish morph tags on the target side. The tags were specified within the aligned sentences as

morphemes preceded by a plus symbol, and trained on a phrase-based system. Though, improvement over the plain phrase-based baseline was gained only after augmenting the generation process with morphotactical knowledge.

The presentation of the factored SMT models (Koehn and Hoang, 2007) describes experiments for translating from English to German, Spanish and Czech. The experiments, apart from POS tags, used morphological analysers with either all or part of the morphology tags (morph). For instance, English-German was improved by using both POS and *morph* factors, English-Spanish performed better with just *morph* factors and English-Czech benefited from partial used of only prepositional, number and gender factors. The factors are added on the output side and exploited with a 7-gram language model.

Table 3.1: Comparison of methods for translating into morphologically rich languages

| from English into | bline% | best% | sets | MT | input feat. | target feat. |
|---|---|---|---|---|---|---|
| Spanish UN03 | 51.1 | 52.4 | 262/13k | IBM4+maxent | (spliced) prn+vrb | - |
| Catalan UN03 | 50.8 | 53.4 | 262/13k | IBM4+maxent | (spliced) prn+vrb | - |
| Russian MTS07 | *77.6 | *91.5 | 1k/1M w | prob model+feat. | POS, morph, syn. | POS, morph, syn. |
| Arabic MTS07 | *31.7 | *73.3 | 1k/1M w | prob model+feat. | POS, morph, syn. | POS, morph, syn. |
| Turkish KO06 | 7.52 | 9.13 | 50/22k | Pharaoh+morphs | POS, lemma | POS, morph |
| German KH07 | 18.04 | 18.22 | 2k/750k | Moses+7gram | - | POS, morph |
| Spanish KH07 | 23.41 | 24.66 | 2k/40k | Moses+7gram | - | morph |
| Czech KH07 | 25.82 | 27.62 | 2k/20k | Moses+7gram | - | CNG,verb,prp |

*methods evaluated by accuracy, set sizes counted in words (not sentences)

A comparison of the methods discussed is shown in table 3.1. We briefly summarise the BLEU score improvements by each of them, the machine translation system they used and the extensions they designed.

## 3.3 Contribution of this project

We have presented recent methods which tried to tackle the problem of unequal morphology when translating between two languages. Few of them try to enrich the translation input by using additional characteristics given by the source language.

This project is strictly focusing on a translation from English to a morphologically richer language. It is going one step further than just using easily acquired information (e.g. English POS or lemmata) (3.2.2). Instead, it focuses on extracting target-specific information from the source sentence context.

We use syntax, not in order to perform complex reordering (3.1), but as a means for getting the missing "morphology" information, depending on the syntactic position of the words in interest. Then, contrary to the methods that added only output features or altered the generation procedure (3.2.2), we used this information in order to augment the input of a factored translation model.

## 3.4 Summary

This chapter briefly described existing approaches which have tried to take syntax and morphology into consideration. We focused on the methods that translate from English to morphologically richer languages and made a comparison of the approaches used and the results achieved. Finally, additions and modifications made by this project were suggested. The next chapter presents the framework and design for proceeding to the experiments.

# Chapter 4

# Experimental framework

Every experiment includes a series of tasks, ranging from the pre-processing of the data to the numerical evaluation of the models. The exact structure of the experiments is shown in this chapter, along with a brief technical overview.

## 4.1 Corpora

All experiments used the version 3 of the English and Greek sections of the Europarl Corpus as a bilingual training source. As a typical procedure in recent SMT tasks, a small part of the data was kept apart (the fourth quarter of the 2000 proceedings), so that it can be used for tuning and evaluation.

### 4.1.1 Basic characteristics

While the initial effort included the entire preparation of the English-Greek data, based on the raw Europarl files from scratch, there was insufficient knowledge on the selection of the test sets in a way that they would mimick the standardised test-sets provided for the WMT07 SMT task. Therefore, we resorted to the aligned ver. 3.1b, which was kindly prepared for us [1].

The final training set consists of approximately 536,000 sentences aligned into pairs. After an essential clean-up, only 440,000 sentence pairs were usable, probably due to issues of incompatible tokenisation and several minor alignment issues, not captured by the algorithms used. As the data loss seems to be quite high, re-adaptation of the preprocessing algorithms to the data would be useful but was not performed due

[1] The sentence-aligned version 3.1b of the English-Greek Europarl section was prepared by the research assistant Josh Schroeder

to time restrictions and since it was not directly connected with the actual aims of the project.

The three test-sets (dev2006, devtest2006, test2007) consist of 2,000 selected Europarl sentences each, with no particular discourse connection between them. The first set (dev2006) was used for the tuning process as well. All three sets seem to share the same language with the training data, mainly political speech referring to the internal workings of the European Union, so the task will be assumed as a single-domain approach.

## 4.2 Software and design

The experiments were performed by following the experimental framework designed and used within the SMT Group of the Edinburgh University. Its main purpose is to incorporate the whole process of creating and evaluating an SMT model in a single script, by executing the essential processes, transferring the data from the one to the other, following the dependencies between them. The use of this script also gave the possibility of constructing and evaluating models of a large size, since many of the tasks and the sub-tasks were parallelised, when possible. As experiments on similar sets of Data (Europarl Corpus) have been already implemented given the current script, its adaptation to the requirements of the project was straightforward. In particular, each

Figure 4.1: Diagram of the experiment process (source: statmt.org/experiment)

experiment consists of the following processes:

- **Corpus preprocessing**: At first, the data need to be tokenised and cleaned-up, verifying the sentence alignments, adding the essential factors and switching all text to the lowercase, as this is essential for avoiding duplicate probability estimation of differently cased instances of the same words.

- **Creating the language model**: The target side of the training corpus was used for building the target language model, which is essential during decoding. After the basic preprocessing, the n-gram model was trained by using the SRILM toolkit (Stolcke, 2002). The same target language model was reused for all experiments which involved factors on the input side, since this wouldn't affect the

application of the output n-gram model. Otherwise, the model was re-trained by using factors, a capability that is being supported by the latest version of SRILM.

- **Translation Model training**: As part of the training task, the GIZA++ tool is used, for acquiring the phrase-pairs. As it has been explained earlier (section 2.1.1, page 6), the lexical alignment process is run in both directions of the translation process and the symmetrisation of the outcome leads to the phrase extraction and learns the lexical translation. Therafter, the extracted phrases are used for training a reordering table, while combined with the set of the lexical translation, all phrases are scored, giving the phrase translation table, containing the probabilities each phrase to be translated into another phrase, as it has been described in the theory of the phrase-based models. When factors are used, the system also builds a generation table, so as to indicate how the multiple translation levels are combined into the surface form. Finally, as an outcome of the training process, the three tables (phrase, reordering and generation) are the essentials for performing the decoding and consequently passed to the next steps

- **Tuning**: The use of a log-linear model for the factored SMT model (equation 2.5, page 10), needs the weights $\lambda_i$ to be determined. The tuning script uses the repetitive process of *maximum error rate training* (MERT) on a tuning set. This way, the decoding procedure is repeated with adapted parameters in order to adjust the weights, so that the achieved BLEU score is maximised. Every iteration is run with a new parameter setting, n-best lists are generated and merged, and the optimal parameters of the iteration are recorded. The iterations stop as soon as the optimal BLEU score converges. Since the algorithm cannot perform an exhaustive search, even for a small number of features, the results of many tuning processes on the same set may vary slightly.

- **Testing and evaluation**: as the translation model has been fully built and the decoding parameters are set, the testing and evaluation task proceeds with decoding the test three test sets. As happens with tuning, the result is compared with a reference tuning translation and a BLEU and a NIST score is given for each of the sets.

The vast majority of the scripts are coded in Perl. The exact experiment sequence, along with the data dependencies and the parallelisation achieved, are depicted in diagram 4.1

### 4.2.1 Language tools

- **English POS tagging**: For the English text part-of speech tagging, we used the rule-based tagger developed by Eric Brill (Brill, 1992) on v1.14. The Brill tagger is based on acquiring rules and tags and was chosen since it gives an accuracy comparable to stochastic taggers and it was for years the state-of-the-art in POS tagging.

- **English syntax parser**: For getting the syntax tree of the source sentence, the latest version of M. Collins' parser (Collins, 1997) was used. The parser uses a generative model of lexicalised context-free grammar.

- **Greek tools**: For the preprocessing and annotation of the Greek data, Ellogon (Petasis et al., 2003) tool was used, along with the associated modules. For POS tagging, it includes a Greek modification of Brill's tagger (Petasis et al., 1999), a word/sentence tokeniser and a module for morphology look-up (Petasis et al., 2001).

### 4.2.2 Factorisation

Since the framework of the experiments was pretty much defined, the main coding effort of the project was concentrated on adding factors on the corpus. All code was written in Python and data were fed to the rest of the script via intermediate files. The exact factorisation process was in focus, during most of the project and will be presented in the next chapter, focusing on the methods that were employed .

### 4.2.3 Evaluation tasks

Additionally to the metric evaluation described to the above script, efforts were made in order to gain conclusions from the translation outcome. A simple script for *pairewise bootstrap sampling* (section 2.3.3, page 13) was implemented to compare each set with the baseline, in terms of defining the statistical significance of the comparison. Further manual error analysis (Vilar et al., 2006) was performed on the test results, in order to identify improvements made within each hypothesis, when necessary. Due to the fact this manual task was pretty time consuming (about one minute per sentence), it took place only on 60 sentences per set (360 sentences per experiment). Of course, the comparison was made on the same 360 sentences of every model. In the last step,

14 annotators were asked to judge the adequacy and the fluency on the outcome of 4 sample models.

## 4.3 Technical aspects of the experiments execution

Due to the size of the training data, all experiments were quite demanding, in terms of computational needs. Several tasks needed a lot of RAM, mainly the ones which dealt with the phrase tables (building, filtering, decoding and tuning). The size of a phrase translation table, varied from 200 MB to 4 GB for each factor level, depending on the experiment. Consequently, the experiments were run in a Sun Grid Engine parallel environment, which provided a lot of RAM (2-8 GB). Most of the phrase tables needed to be binarised, in order to allow decoding without being loaded into memory. Part of the framework used, included the parallelisation of many processes (mainly decoding and factorisation), by splitting the input in many files, processing them in many parallel tasks and then gathering the output.

The total processing time for an experiment varied between 2 days and one week, depending on the parameters and the availability of CPU. Several Grid Engine technical problems, which were not within our scope of responsibility, delayed many of the experiments of the project, causing unwilling crashes for period of time. That was a reason for having not much flexibility in examining many of the possible options for verifying our assumptions.

## 4.4 Summary

This chapter examined the technical details and experimental design of the project. The next chapter is presenting the implementation in more detail.

# Chapter 5

# Methods for enriching input

Having identified the need for enriching the English text with additional linguistic information in this section we proceed with further details on methods than can be useful.

## 5.1 Defining the problem

As a part of defining the problem in a better way, we focus on detailed linguistic and structural details that justify the need for our efforts. This will enable us to easily take design decisions on how to develop the possible solutions.

### 5.1.1 Data analysis

As an effort to verify whether the used language pair and data are suitable for the purpose of the project and further focus on the problem, a basic corpus analysis was performed. From the comparison of the two languages (table 5.1), we can draw the following points:

- The count of distinct Greek word forms is 2.5 times the count of distinct English word forms, which can be explained by the richer morphology. Even if all terms in both languages are separated according to their part-of-speech role, the proportion is still quite high, about 2:1. Unfortunately, a parallel comparison of lemmata was not possible since there was not a comparable lemmatisation method for both languages.
- English language uses 9% more words per sentence in average; this is an indication of structural issues (eg pronoun + verb) and phenomena which use more

Table 5.1: Comparative analysis of the Greek-English language

| | English | Greek |
|---|---|---|
| number of sentences | 440 082 | 440 082 |
| number of tokens | 11 613 530 | 10 574 397 |
| number of characters | 63 629 205 | 57 601 542 |
| avg sentence length | 26.4 | 24.0 |
| avg word length | 5.5 | 5.45 |
| distinct word types | 58 159 | 135 000 |
| distinct word types as POS | 70 503 | 138 893 |
| SRILM perplexity | 56.8797 | 62.4458 |

phrases to express the same meaning.

- Using POS tags along with the word surface, in order to disambiguate the English terms, gave 12,000 more distinct words. This refers to the fact that many words (e.g. *stop* or *sink*) may function both as verbs, nouns etc, which obviously indicates high lexical ambiguity, affecting the 21% of the distinct source words.

- Measuring the perplexity on the language models of the corpora that have been built with the same parameters (5-ordered n-grams with Knesser-Ney smoothing (Stolcke, 2002)), shows that the Greek language model has a higher perplexity. This is another indication of the fact that Greek uses more complicated structures to infer the same meanings.

These findings give a good motivate for using the data for the particular purpose that has been described. More detailed analysis on the data will be given per case.

### 5.1.2 Problem identification

When examining a sentence pair of two languages of the training corpus, it is apparent that for many words/phrases which in English appear most usually in the same form, the corresponding Greek terms would appear inflected in many different ways. On a single word-based probabilistic level, it is then obvious that for one specific English word $e$ the probability $p(e|f)$ of it being translated by a word $f$ (formula 2.1) decreases, as the translation candidates increase, often making the decisions quite uncertain.

One of the main aspects that signify that a formed sentence is fluent, is known as the prerequisite of *agreement*, which reflects the need of correspondence on gender, case, number and person within a sentence. The exact rules of agreement are language-dependent and are closely linked to the morphological structure of the language.

The core implementations of SMT can so far deal with these problems in two ways:

1. The basic SMT interpretation of the Bayes noisy channel (formula 2.3, 7), uses the target language model as a factor in the argument maximisation function. This language model has been trained on pieces of grammatically correct text, and would therefore give a good probability for word sequences that are likely to occur in a sentence, while it would penalise ungrammatical or badly ordered formations.

2. Meanwhile, in phrase-based SMT models (chapter 2.1, page 5) probabilities are assigned in sentence chunks. This can resolve phenomena where the English side uses more than one words to describe what is denoted on the target side by one morphologically inflected term.

Though, with respect to these methods, the problem becomes clear when agreement needs to be applied on a sentence length which exceeds the "n-gram frame" of the target language model and the chunk being translated is not a seen event at its whole length. Three common aspects of agreement are as following:

#### 5.1.2.1 Noun cases

Noun cases are know as the most challenging difference in language pairs between case-less languages (e.g. English, French, Spanish, Swedish, Italian) and the ones who do use cases (e.g. German, Greek etc). The case is mainly defined by the syntactic part of speech that each noun has, given very specific rules. Nominative case is used to define the nouns which are the subject of the sentence, accusative shows usually the direct object of the verbs and dative case refers to the indirect object of bi-transitive verbs. Finally, vocative addresses a speech to a person.

#### 5.1.2.2 Verb conjugation

This term refers to the fact that the Greek verbs are inflected according to their use in the sentence. For most European languages conjugation includes characteristics

such as the person, number, mood, tense and voice (Arabic verbs may also include the gender). English verb itself rarely includes information about the person, the mood and the number, while our target language verbs do. It seems that this information is mainly inferred by adjacent words; the leading pronoun determines the person and the number and usually the subject of the hierarchically higher clause is inferred as a subject of the verbs which are in subclauses and lack a clear person definition.

Given this fact, the use of phrase-based models has resolved the issue of conjugation in simple cases (e.g. 'pronoun+verb'). Though, in more complicated cases, for example when in the source language the subject of the sentence has a distance from the verb, and the target language verb includes the person and number information by omitting the subject, language models usually lead to the verb inflection which is the most often during training, with no respect to the syntactic context.

#### 5.1.2.3 Gender

> Mr President, my apologies for interrupting you (...). In the Greek text Mrs Jutta Haug is referred to as a male rapporteur and Mr Markus Ferber is referred to as a female rapporteur. I should imagine that this double sex change in a day and without surgical intervention is the first of its kind and it deserves a mention in the Guinness Book of Records, unless of course the necessary corrections are made and all changes of sex avoided.
>
> *I. Marinos, Member of European parliament, 15/12/2000*

Several grammatical terms within a sentence have to be consistent in their gender. Gender issues mainly apply to adjectives, articles, pronouns and copula verbs, which usually have to comply with the noun or the dominating pronoun. Most cases, when the gendered predicate is close to the verb, are covered by the phrase-based model, but others with a lot of adjectives in several positions, are hard to be resolved.

Considering English as a source language, a gender value is mostly applied on sentences referring to third-person pronouns and proper names. Adjectives do not have a gender at all and this is the case for the vast majority of the nouns, which are usually considered neutral or they inherit the gender of the named entity they refer to (without this having an impact on the appearance of the word). However, the Greek gender is shown in the morphology of every adjective and noun and moreover, nouns

Table 5.2: The gender distribution in the Greek dataset

| gender | total | article | adject. | noun | pron. |
|---|---|---|---|---|---|
| male | 539 435 | 70 684 | 131 085 | 308 289 | 66 932 |
| female | 2 560 754 | 392 625 | 492 038 | 1 182 146 | 577 812 |
| neutral | 1 735 068 | 196 578 | 686 189 | 482 492 | 432 071 |

that refer to objects do not have a specific gender (e.g. a chair is female, the computer male and the moon is neutral).

This rather uncomfortable "incompatibility" has obvious impact on human translations as well, when not enough discourse information is available to disambiguate the gender of people or professions. This is the case that the above quote illustrates.
Table 5.2 shows the distribution of the gender among the various parts of speech of the Greek Corpus, as annotated by the Hellenic Morph analyser of the Ellogon tool.

#### 5.1.2.4 Other phenomena

It is quite likely that several additional linguistic phenomena may affect the sentence agreement (e.g. the mood of the verbs). Though, we will currently focus on these three aspects since they are indicative enough of the problem investigated. As the problems have been hypothetically identified, the next sections proceed with explaining the solutions proposed for each of these.

## 5.2 Discovering noun cases

It has been clear, so far, that our intention is to enrich the translation input with information in respect of the behaviour of the target language. As the first task was determined to cover the issue of noun cases, the main goal is to treat the nouns of the English text as if they were part of a Greek text, trying to identify what cases they would have then.

The approach followed takes advantage of syntax. Since English, as morphologically poor language, usually follows a fixed word order (subject-verb-object), a syntax parser can easily identify the subject and the object of the sentence, for instance. If

such annotation is taken into consideration, a factored word model would be easily trained to map the word-case pair to the correct inflection of the target noun. Given the agreement restriction, all words that accompany the noun (adjectives, articles etc.) must follow the case of the noun, so their "potential" case needs to be identified as well.

For this purpose we used the Michael Collins' syntax parser, in order to produce a syntax tree out of every English sentence. With a simple object-oriented python script, the trees were parsed depth-first and the cases were identified within particular "sub-tree patterns" which were specified as rules. Practically, the script used the sequence of the nodes in the tree to identify the syntactic constituency of each one:

- **Nominative**: The nodes for which there was a clear indication that they function as a subject of the verb, were tagged to be of a nominative case. Also, the arguments of copulas are in nominative case, whether the copulas are verbs or prepositions

- **Accusative**: The nodes which were an object of a transitive verb were tagged to be of an accusative case. This was also the case for most of the prepositional phrases. Since in modern Greek the **dative** case has been replaced by a prepositional phrase using accusative (as is the formation in English), we didn't mark distinctively the dative case.

- **Genitive**: The genitive case was tagged on possessive clauses (*'s*) or on prepositional phrases of the same meaning (mainly introduced by *of*).

- **Vocative**: This case was completely ignored, for two reasons. At first, the use of vocative case cannot be clearly identified by tree structures, since it mainly resides in noun-phrases which our parser tends to attach as a subject to a verb. On the other side, the Greek vocative is in most cases identical to the nominative, plus it never expands to other subtrees or n-grams longer than three words.

In particular, in each experiment, after the whole tree has been parsed and all tags are added as necessary, the leaves of the tree keep the annotation of their parent node only if they function as a part-of-speech which can support a case. In the current approach, a case was allowed for pronouns, proper nouns, nouns, adjectives, articles and participles.

To make things more clear, an example can be seen in figure 5.1. At first, the algorithm identifies the subtree "S-(NPB-VP)" and the *nominative* tag is applied on

the NPB node, so that it is transferred to the word “We” which will have it assigned, since a preposition can have a case. The example of accusative shows how cases get transferred to nested subtrees. In practice, they are recursively transferred to every underlying noun phrase (NP) but not to clauses that don’t need this information (e.g. prepositional phrases).

Figure 5.1: Applying noun cases on an English syntax tree

At this point, it must be noted that many of the details of this particular annotation are just a result of an assumption on how the translation can be eased by their use. A full experimental research would need to include, if possible, an exhaustive combinatorial search of many parametrisations, since the way a factored phrase-based translation model generalises the corresponding probabilities cannot be predetermined.

## 5.3 Verb persons

This task is focused on identifying the person of a verb and adding this piece of linguistic information as a tag. It is obvious that syntactic knowledge is essential for this task too, so we are once again post-processing the outcome of the syntax parser.

The script which implements these rules was a modification of the previously described tree analyser. As the script parses the tree top-down, on every level, it looks for two discrete nodes which, somewhere in their children, include the verb and the corresponding subject. Consequently, the node which contains the subject is searched recursively until a subject is found. Then, the person is identified and the tag is assigned to the node which contains the verb, which recursively bequeath this tag to the nested subtree.

It needs to be mentioned that both the type and the order of these “two discrete nodes” were taken into account. This was to avoid wrong assignments, since for example a noun phrase may contain a subject only if it precedes the corresponding verb phrase, otherwise it may be an object. In specific, this parse applies to pairs preceded by noun phrases and followed by either a verb phrase, a secondary clause containing a referring “wh-sentence”. We captured also cases of prepositional phrases introducing verb-like expressions, mainly starting with ‘for’ and ‘to’ (e.g. “*he is the one to (he)read the book*”, “*I chose him for (he)playing the role*”).

For the subject selection, the following rules were applied:

- The verb person is directly connected to the subject of the sentence and in most cases it is directly inferred by a personal pronoun (I, you etc). Therefore, since this is usually the case, when a pronoun existed, it was directly used as a tag.
- All pronouns in a different case (e.g. '*them, myself*) were nominised before being used as a tag.
- When there is not a pronoun as a head of the sentence, but a single noun, then this is obviously the third person. The POS tag of this noun is then used to identify if it is in plural or singular number. In this case we tried to exclude nouns which despite being in singular, take a verb in plural.
- The verbs do not need to know the gender of the subject. Therefore, all three genders that are given by the third person pronouns were reduced to one.

Figure 5.2: Applying verb person tags on an English syntax tree

In figure 5.2 we can see an example of how the person tag is extracted from the subject of the sentence and gets passed to the relative clause. In particular, as the algorithm parses the syntax tree, it identifies the sub-tree which has NP-A as a head and includes the WHNP node. Consequently, it recursively browses the preceding NPB so as to get the subject of the sentence. The word "aspects" is found, which has a POS tag that shows it is a plural noun. Therefore, we consider the subject to be of the third person in plural (tagged by *they*) which is recursively passed to the children of the head node.

## 5.4 Efforts to resolve the gender issue

In the previous sections, the main effort was to try and directly extract the "missing" linguistic information from nearby words, having in mind how specific language chunks (i.e. noun phrases, verb phrases) would behave if they were in the target language. Unfortunately this does not seem to be applicable for resolving the gender. This is because gender information in English is mostly incompatible with the one in Greek, or sometimes completely nonexistent.

Since we are focusing on preprocessing, it is quite hard, given the English word, to make any prediction on the gender of the target word. Consider nouns referring to inanimate objects, which define how their adjectives should be inflected: such an approach would require a word-based translation (or a lexical look-up), in combination with a Greek morphological tool, in order to identify that kind of "potential" gender which would then be used as a tag for the adjective. However, this drives the translation task far from the benefits of a phrase-based model, since the same noun may have multiple translations with different genders, or be part of collocation.

We tried to reform this idea, in order to avoid a separate level of a possibly noisy word-based translation. Let's concentrate on a subset of the whole problem, the adjectives. The gender training can be possibly captured within a factored translation model if go one step back on the agreement prerequisite: an adjective would need to know (as a factor) the gender of the noun it refers to. But since this is not possible, we will give as a factor the whole referring noun instead. As it will be seen later on the experiments, this noun can be mapped to a gender with target side factors, hoping that this would lead to the generation of a properly inflected adjective.

Consequently, in terms of implementation, it is only needed for every adjective to get the noun it is referring to. Again, this script was based on a generalisation of the previous methods. At first, there was the "easy" task of resolving the adjectives which exist next to the noun, almost always in the same noun phrase. Using POS tags, the noun was underlined and noted as the leading noun of the noun phrase, added as a tag to the node and inherited by all the nearby adjectives. But since the actual challenge is when there is a detached adjective, e.g. functioning as the argument of a copula/verb, the leading noun had to be generalised over the sentence.

This was not difficult given the top-bottom tree traversal: The leading noun of each noun phrase was passed to all phrases that were in lower hierarchy. If these phrases have their own leading noun, then the inherited tag will be overwritten when they are traversed, along with all the nested noun phrases. Of course, this approach supposes that our syntax parser can properly resolve the hierarchy of the nested phrases, which was not always the case.

Figure 5.3: Resolving the noun reference to the verb arguments

Let's see figure 5.4: First, (loop 1, step 1) starting from the node S, the algorithm

checks the node patterns of its children and identifies the pair NP-A, VP so it knows it needs a subject. So (loop 1, step 2) it looks recursively into NP-A, so as to find the noun *issue* into the first NPB. The tag *issue* is given (step 3) as a tag to the S node and, as it happens when a tag is given, that tag is inherited (step 4) by all the children of the sentence node. The top-bottom traversal continues on the first children of the S node (loop 2, 3...). When it finds NPBs, it locates the dominant noun and gives that as a tag to adjectives and articles in that local NPB scope.

If annotation has succeeded, so far, all subclauses have their adjectives annotated with the referring noun. Though, what happens when an abstract pronoun of the second subclause (e.g. *this*, *they*) refers to a noun mentioned earlier? This problem (known as *pronominal anaphora resolution*) seems to bear more relationship to word sense disambiguation (WSD) efforts. Several algorithms and existing tools were investigated, but most of those found, focus on resolving pronouns that refer to named entities; however, we are currently focusing on the inanimate nouns. Finally we just implemented basic rules inspired from the ones used for the third person pronominal resolution in a rule-based MT system (Mitamura et al., 2002):

- As a candidate antecedent for the pronoun only nouns, units, tags or conjoined NPs were allowed.
- The antecedent must precede the pronoun, if they are in the same sentence. In this implementation, in conjoined sentences, only the first one was allowed to generalise its leading noun to the whole discourse.
- If the antecedent is a conjoined NP, they must be conjoined with *and* or *or*.
- Antecedents that were a part of a co-ordination were pruned
- The pronoun and the candidate antecedent had to agree in number (when a conjunction was identified, it was considered plural).

The rest two rules, which required resolving of the verb arguments and the objects of the prepositional phrases, were ignored. Though, prepositional phrases and wh-sentences were explored as previously.

Pronominal anaphora resolution normally takes place over discourses created by many consequent sentences. Though, even if inter-sentence anaphora resolution succeeded on the training set (where sentences follow each other in their normal order) that would be impossible on the test since they consist of randomly selected sentences.

While this shows limitations of the evaluation system, it had been decided to stick with the standardised evaluation sets. Therefore, we hope that such technique may be beneficial at least for the long sentences of the corpus, which include many conjoined sentences.

Figure 5.4: Simple pronominal anaphora resolution for connected sentences

A benefit of the pronominal resolution can be seen in figure 5.4. When we have conjoined sentences, the assignment of the noun tag is forced to exceed the current scope, for the first sentence, so as to be available for any further pronoun resolutions. When the second sentence was parsed, its subject was identified to be *they*. Since *they* matches the number of the already given tag (*buses*), the pronoun will not overwrite the previously assigned noun, letting the rest of the sentence to use the word *buses* as a tag.

During the implementation, it emerged that the Greek adjectives also require an agreement on the number of the noun. That was easy to be acquired from the leading noun, just by using its POS tag.

## 5.5 Limitations and constraints on implementation

The approaches described are obviously based on rules, which had to be specified manually, by browsing through a limited set of sentences. Therefore, their applicability is quite limited, since there is nothing to ensure that there cover every possible linguistic phenomenon that may show up later. We assume that the samples examined are indicative of the whole text, enough to show some improvement. Also, most rules did not generalise for unseen subtrees, just to avoid erroneous factor assignments.

The limitations that emerge from the use of the M. Collins parser need to be mentioned, at this point. At first, such a probabilistic parser cannot always be able to fully disambiguate all possible sub-tree combinations. Its application needed a lot of preprocessing to tackle with its incapability to parse (some of the very few) Unicode characters.

Quite serious was the fact that it completely failed to create a parse tree for many long sentences, usually longer than 50 words. The failure was estimated to be at about 1% for the training set (which is ignorable due to the training parameters described

later) while it was significantly higher (about 4%) for the test sets. It was verified that it was not a sporadic problem of memory management, nor directly connected to the length of the sentence, but originating to some incompatible sub-trees. In order to overcome this problem, the unparsed sentences were trimmed to a length of 50 words and then were reparsed, so there were tags assigned only on the first 50 words of every sentence, since this would most likely correctly produce tags for part of the sentence, but this is deinitely better than no tags at all, for a whole sentence. This problem was a harsh limitation, since the concepts used are particularly benefitial for long sentences.

## 5.6 Conclusion

This chapter analysed the linguistic differences between the two languages of the data, by comparing several basic characteristics. Also, it presented the methods and the implementation details for enriching linguistic information on the English side of the corpus. In the next chapter we will describe and comment on the performance of these methods, focusing on the outcome of the experiments.

# Chapter 6

# Experiments and results

Having presented the theoretical background and the methods for acquiring the essential linguistic information on the given dataset pair, it is time to examine their effect on the actual task of SMT. This chapter will present the exact parameters each experiment was run with and the related results. It also includes the outcome of the efforts to explain the gains or drawbacks of each experiment.

## 6.1 Baseline experiment

The baseline was an English to Greek translation, trained on all 440,084 pairs of sentences, after pruning the sentences which were longer than 60 words, since they would increase complexity. An *n*-gram of order 5 was used for the target language model. Giza++ was run in three parts and its two directions were directed to the *grow-diag-final-and* symmetrisation method (Koehn et al., 2003) to obtain word alignments. Lexicalised reordering was also used. The baseline included no factors.

The baseline parameters were kept intact for all the experiments, apart from the input, output and generation factors. The reordering was also kept on a word-level.

### 6.1.1 Baseline performance

It is obvious that the Greek translation performs comparatively worse (18.09% average BLEU score [1]) than models translating into other languages, made on the same system with similar SMT parameters. This obviously has first to do with the smaller size of

[1] All experiments were tuned (MERT) on the dev2006 set. Therefore, despite we present dev2006 in all tables of results, it is not taken into consideration for calculating the average of the metrics for every model

Table 6.1: BLEU and NIST scores of the baseline system

| | BLEU | NIST |
|---|---|---|
| dev2006 | 17.90 | 5.215 |
| detest2006 | 18.05 | 5.218 |
| test2007 | 18.13 | 5.279 |
| average | 18.09 | 5.249 |

the usable data, which is just one third of the size of standardised (WMT07) Europarl tasks, which train on 1,3 million sentences. Then, the difference is also affected by the linguistic differences of the two languages. Since our effort is to create models that perform better than the baseline, it is useful to perform an error analysis on this basic translation.

#### 6.1.1.1 Reference quality

During a manual evaluation, the first thing to point out, that can be accounted as an additional reason for the low score, is the quality of the reference text. It appears that human translators prefer to rephrase many chunks of the input text, possibly in order to better reflect the meaning or to stick to the way a political speech would be expressed in Greek (which generally includes more formalisms and old language). For usual introductory or connective expressions, the phrase-based model performs quite well, in most cases avoiding a dummy word to word translation but also preserving the meaning of the sentence (not essentially agreeing with the reference text). Though, as translators occasionally choose to do unnecessary reorderings and phrase replacements, even correct sentences can be given a bad score. The bad score is reflected in simple decoding, but also affects the overall performance due to the fact that tuning is based on BLEU scores.

Figure 6.1: Example of the rephrasings noted in reference translations

For example, in figure 6.1 where an accurate back-translation is shown, the human translator reordered and translated the phrase “we are at least as responsible for” as “a duty of ours is to” and substituted the “make progress” with “go on”. Apart from

the few grammatical and vocabulary mistakes, the machine-translated sentence bears more relationship to the original one.

#### 6.1.1.2 Identifying mistakes

As seen in the previous excerpt, grammatical mistakes were notable in most sentences, showing the need for making an effort to improve that. For getting a whole glimpse of the range of the errors, we performed an error analysis as described by Vilar et al. (2006), essentially modified so as to better categorise the grammatical errors. The class "incorrect forms" was futher divided so as discriminate errors on verb conjugation (person, mood, voice), gender, part-of-speech and noun cases. An extra class was added to define when a sentence had an opposite meaning than it should.

Figure 6.2: Error analysis on the baseline system

Based on this categorisation, the average distribution of errors, for all three test sets, can be seen in figure 6.2. Some points on the shoen analysis could be noted:

- **Missing words**: There was a considerable amount of missing words, which are about equally divided into *content words* (words with meaning, such as nouns, verbs etc) and *filler words* (supplementary words such as prepositions, articles etc.). Missing content words are usually due to the contribution of the target language model: Ungrammatical, wrongly translated or untranslated words produced by the translation components create non fluent $n$-grams which most usually do not comply with the target language model. Therefore, skipping these words leads to a higher sentence probability in overall and this is determined by the weights given during the MERT tuning process. Similarly, missing filler words has to do with the incapability to indicate the correct sequence of words for combining translated phrases. There is a hope that reducing the ungrammatical words based on the methods presented, will reduce the number of the skipped words.

- **Extra words**: If phrases give better probability when translated as a whole, than in parts, phrasal chunks may introduce unwanted expressions. While both this and the previous problem are related to sparse data and the target language

model, it remains interesting to follow how their count is affected by introducing factors, in the next experiments.

- **Reordering**: It has been chosen not to pay much attention to reordering, therefore we just counted both phrasal and lexical reordering in one class. Reordering errors were very few and didn't affect much the final results.

- **Incorrect form**: This class is the most interesting one, as it mainly reflects grammatical errors on agreement and similar rules. It also appears that almost half of the translation errors are in this category. Obviously, this indicates that improving this kind of errors is quite important for the overall outcome. We also have to note that the three most important error categories are related with the verb person, the noun cases and the gender, giving ground to further emphasising on this problems.

## 6.2 Adding POS tags

Part of the "introductory" experiments involved the simple use of POS tags, in order to augment the source words. A standard tagset with 41 POS tags was used. Three experiments were performed, mainly aiming to observe the factorisation capabilities.

Table 6.2: BLEU and NIST score for experiments using English POS tags

BLEU

| | dev2006 | | devtest2006 | | test2007 | | avg |
|---|---|---|---|---|---|---|---|
| Baseline | 17.90 | | 18.05 | | 18.13 | | 18.09 |
| word+POS→word | 18.06 | $\sim 88\%$ | 18.27 | $> 95\%$ | 18.19 | $\sim 61\%$ | 18.23 |
| POS (no reord.) | 17.76 | $\sim 83\%$ | 17.95 | $\sim 79\%$ | 17.84 | $< 98\%$ | 17.90 |

NIST

| | dev2006 | | devtest2006 | | test2007 | | avg |
|---|---|---|---|---|---|---|---|
| Baseline | 5.216 | | 5.218 | | 5.279 | | 5.249 |
| word+POS→word | 5.245 | $> 95\%$ | 5.271 | $> 99\%$ | 5.311 | $> 95\%$ | 5.291 |
| POS (no reord.) | 5.235 | $\sim 80\%$ | 5.242 | $> 90\%$ | 5.281 | $\sim 54\%$ | 5.262 |

### 6.2.1 Using a single translation level

The first experiment was performed by combining the two factor levels before building the translation component. This resulted in one single translation table, which mapped the phrases created by source pairs directly to the output phrases(*word+POS word*) , in a way that every source pair is altogether assigned one probability.

The results of the metrics show some slight improvement, but only one BLEU score is statistically significant, having an improvement of 0.22 . Though, we can safely have a comparison based on the NIST score, which shows an average improvement of 0.042 .

Table 6.3: Error analysis for experiment using English POS tags

| | baseline | w+POS |
|---|---|---|
| **Sense, reord. & lexical choice** | | |
| Missing content words | 8.9% | 7.8% |
| Missing filler words | 10.8% | 8.4% |
| Local range order | 4.4% | 1.7% |
| Long range order | 4.4% | 1.8% |
| Wrong lexical choice | 15.7% | 14.7% |
| **Word form errors** | | |
| verb person | 18.9% | 21.3% |
| gender | 8.5% | 11.8% |
| pos | 4.2% | 2.9% |
| noun case VP | 4.4% | 4.3% |
| noun case PP | 5.7% | 8.9% |
| mood | 0.6% | 3.2% |
| tense | 0.6% | 1.4% |
| voice | 0.8% | 1.4% |
| **Various errors** | | |
| extra words | 8.7% | 5.8% |
| unknown words | 1.3% | 2.3% |
| punctuation | 0.4% | 0.3% |
| negative meaning | 1.5% | 2.0% |

The manual evaluation (table 6.3) reveals a slight improvement in the number of errors, especially to those connected to the fluency of the text. Missing words have been

reduced. Meanwhile, this model also fails to produce grammatically correct sentences.

While the improvement is quite small, the improvement can be explained by the fact that in the English side of the corpus, more than 20% of the words are ambiguous, when considering what part-of-speech they consist (chapter 5.1.1, page 27). While much of the ambiguity usually refers to similar POS tags (eg noun and proper name), the improvement can be obviously attributed to more serious ambiguities (eg noun and verb).

### 6.2.2 Using two translation components

The second experiment using POS tags involved building two translation tables. The first one mapped the input word surface to the output word surface and the second one mapped the input POS tags to the output word surface as well. Due to the very low probability of $p(word_{transl}|pos_{source})$, the decoder prematurely pruned most of the translation candidates (see chapter 2.1.3, page 8). This way there was practically no translation produced in the output.

Using a single translation table practically treats every word with its factor as a spliced chunk, so the data get more sparse. An alternative for using two translation levels, was to use a system with the main translation component as previously ($word + pos \rightarrow word$), but having a backoff table ($word \rightarrow word$), which would treat cases when a word has to be decoded with a factor which it hasn't been trained with. Unfortunately the software did not allow us perform any decoding on this basis, possibly due to a bug. It was only possible to combine the POS augmented phrase table ($word + pos \rightarrow word$) with the baseline one ($word \rightarrow word$), into a model which treated the two tables as two separate translation components with constant weights, which didn't seem to be any useful.

### 6.2.3 The effect of reordering

Despite performing experiments on reordering was not in the priorities of the project, we performed an experiment to evaluate the contribution of the lexicalised reordering on the case of using POS tags. Therefore, we removed the word to word lexicalised reordering. This gave a 0.19 BLEU score decrease, performing even worse than the baseline. This confirmed that lexicalised reordering was definitely useful and therefore will be used in the following experiments.

## 6.3 Using factors for noun cases

While POS tags proved to be useful, it is clear that not much linguistic information are taken into consideration this way. Therefore, we proceed with the more linguistically motivated experiment focusing on noun cases.

Table 6.5: BLEU and NIST score for experiments using noun cases

BLEU

| | dev2006 | | devtest2006 | | test2007 | | avg |
|---|---|---|---|---|---|---|---|
| Baseline | 17.90 | | 18.05 | | 18.13 | | 18.09 |
| word+case→word | 17.69 | ∼ 89% | 17.58 | ∼ 49% | 17.94 | < 91% | 17.76 |
| w→w, case→case, w+case→w | 1.29 | < 99% | 1.47 | < 99% | 1.26 | < 99% | 1.37 |
| w→w, case→case, case←word | 13.84 | < 99% | 13.78 | < 95% | 13.97 | < 99% | 13.88 |

NIST

| | dev2006 | | devtest2006 | | test2007 | | avg |
|---|---|---|---|---|---|---|---|
| Baseline | 5.216 | | 5.218 | | 5.279 | | 5.249 |
| word+case→word | 5.205 | ∼ 67% | 5.160 | < 99% | 5.250 | < 93% | 5.205 |
| w→w, case→case, case←w | 4.697 | < 99% | 4.679 | < 99% | 4.732 | < 99% | 4.706 |

### 6.3.1 Using only input factors

The tags for the noun cases were generated on the English text, as explained in 5.2 (page 31). Factors were obtained for pronouns, proper nouns, nouns, adjectives, articles and participles, according to their syntactic roles.

The initial plan included two experiments, concerning on whether prepositional phrases were annotated or not. Both experiments were run in parallel, but due to insufficient disc space it was impossible to get the result of the experiment excluding the tags on prepositional phrases in the time allocated for this task. Therefore, priority was given to the experiment that tagged both main sentence constituents and prepositional phrases, since this way the model was trained on the inflected forms of most of the caseable words of the text, giving a hope for less sparse data.

#### 6.3.1.1 Metric results

Not all of the results are significantly comparable with the baseline, given the pairwise bootstrap comparison. However, judging from the test sets which are significantly

comparable, the translation quality is lower, at about 0.2% BLEU score.

#### 6.3.1.2 Redundancy of noun case tags

The translation outcome is of obviously worse quality than the baseline, showing that the noun case tags did not manage to improve fluency or adequacy. One of the most apparent facts, during the manual error analysis on the produced output, was the increase on the number of the missing content words. As a fair amount of content words are usually nouns, we can assume that the lack of such words is due to a situation of sparse data, as it has been identified previously. Since our model is trained on spliced *word+factor* units, when a noun has only been trained e.g. only as accusative, then the decoder will fail to produce any translation for this word.

The cause for this can be further attributed in the distribution of the distinct word forms in the Greek nouns and adjectives: only the male ones (along with some very few female ones) have a distinct word form for the accusative and the nominative. Meanwhile, nominative and accusative case for the female nouns is usually differentiated by the article, while the neutral ones do not have a distinct article either. This obviously fragments the target word-surface probability into more than one spliced word units; while for male nouns or frequent words of other genders this would not be a problem, for the rest ones which may have happened to be trained with only one case tag, there will be no translation. This would lead either to an untranslated word appearing in the output, or to a missing word, after being penalised by the target language model as not fluent. An example of what has been described can be seen in figure 6.3.

Figure 6.3: The use of case tags depends on the gender of the noun

An effort to reduce this kind of redundancy would include reducing the factors, so as that they only annotate articles, which seem to map better, between the two languages. Unfortunately this experiment was not possible to be executed during the time allocated for this task.

#### 6.3.1.3 Noun phrases or prepositional phrases?

Within the small set of the manual error analysis 6.6, we can see that our effort was somehow effective. The errors due to verb-based noun phrases were reduced at about

Table 6.6: Error analysis for experiment with case factors

| | baseline | cases |
|---|---|---|
| Sense, reord. & lex. choice | | |
| Missing content words | 8.9% | 13.8% |
| Missing filler words | 10.8% | 9.6% |
| Local range order | 4.4% | 4.6% |
| Long range order | 4.4% | 6.1% |
| Wrong lexical choice | 15.7% | 14.7% |
| **Word form errors** | | |
| verb person | 18.9% | 15.5% |
| gender | 8.5% | 8.0% |
| pos | 4.2% | 2.9% |
| noun case VP | 4.4% | 2.5% |
| noun case PP | 5.7% | 4.2% |
| mood | 0.6% | 1.9% |
| tense | 0.6% | 2.1% |
| voice | 0.8% | 2.3% |
| **Various errors** | | |
| extra words | 8.7% | 7.8% |
| unknown words | 1.3% | 2.7% |
| punctuation | 0.4% | 0.2% |
| negative meaning | 1.5% | 1.0% |

1.4%, while the ones referring to prepositional phrases at 1.7%. The fact that the decrease is not very conclusive has obviously to do with the following facts:

a. As it has been explained, the same experiment makes an effort to model the cases that exist in both noun phrases (as verb constituents) and prepositional phrases. In the latter problem, the tags assigned on the nouns following a preposition were given a tag according to a prediction of the most possible translation of that preposition. It must also be mentioned here that this assumption does not essentially hold, given the fact that many English prepositions can have several possible translations, each of them implying a different Greek noun case. Similarly affected were the phrasal verbs, where it is the verb (and not the preposition) that defines how the prepositional phrase should be introduced.

b. The tree-based rules used for the factorisation, were manually created upon a small set of development data. Even if these rules have been tested in many sentences, there is nothing to verify that these rules can be sufficiently generalised upon the whole test set. Missing rules are noticeable even within the translated sentence and, if further improvement was possible, we would retrain after applying many of those missing rules.

c. Finally, many of the errors were due to the incapability of the syntax parser to indicate the correct hierarchy of the tree nodes, by which we extract the phrasal dependencies. Beyond that, it has also been pointed out (section 5.5, page 38) that the syntax parser was usually unable to handle sentences longer than 50 words, which were only partially parsed. Therefore, the rest of the words didn't have any tags at all, in both training and decoding, which obviously worsened the sparsity of the data.

### 6.3.2 Mapping case factors in both sides

Since the whole framework has been based on factored models, that gives the possibility of using an additional translation component, just for the factors. This additional translation component is based on a separate translation table using equivalent factors on both sides ($case \rightarrow case$) and both translation tables are combined in a log-linear model (chapter 2.2.2, page 10) with the necessarily adjusted weights.

Since this was the first effort including output factors, we experimented using two types of generation. The first one generated the surface word by joining the probabilities of $p(word)$ and $p(case)$, which using the common annotation would be $word + case \rightarrow word$. However, as it can be seen in table 6.5, this method was completely unsuccessful. This is because the probability for generating the $word_m$ for a corpus of n words (for simplicity we are referring to words instead of phrases), would be:

$$p(word_m|case) \approx \frac{count(word_m|case)}{\sum_n^{i=1} count(word_i, case)} \tag{6.1}$$

The size of the denominator leads to so low a probability, that the decoder prunes most of the useful translation candindates. In order to overcome this problem, the decoder was configured to use a probability on the opposite direction ($p(case|word)$) which, since there is a small number of case factors, is calculated in a dissent magnitude. Due to its effectiveness, this generation type (denoted in table 6.5 by $case \leftarrow word$) is chosen to be used for all the following experiments which use factors on both sides.

Additionally to this experiment, there was also an effort to address the dependency of the sparsity on the gender (as explained in 6.3.1.2). Consequently, a model that would take the gender of the output word into consideration ( $word_{source} \rightarrow gender_{target}$ , $case_{source} \rightarrow gender_{target} + case_{target}$ , $word_{source} + word_{target}$ ), such a multiple translation table would require a long decoding process to be decoded and tuned.

#### 6.3.2.1 Acquiring Greek case tags

For this purpose, there was an effort to produce case tags on the Greek side, which would be directly mapped to the English ones. After some research, it became possible to have our data annotated by a Greek morphology tagger, which was able to identify noun-cases based on the morphology of each single word. Two points need to be mentioned here:

a. The morphology tagger was not available as an executable software, since it is not an open-source program. Therefore, the data were kindly prepared on demand, for this project. This did not allow for much flexibility, since all the factorisation process was strictly fixed on the specific piece of data.

b. Tagging was performed based on a lexicon of about 60,000 lemmata. No probabilistic method was used to model $n$-gram sequences and resolve ambiguities. There was not any syntactic information either. This had as a result multiple case-tags to be given to the same token, indifferent of its position in the sentence or any other contextual information. Since it was decided for the experiment to run using only one factor per side, we had to filter the factors. A priority was arbitrarily assigned to each noun case (nominative > accusative > genitive > vocative), and whenever a second case existed for the same word, only the tag with the higher priority was kept. There were also nouns and adjectives with no tags at all.

#### 6.3.2.2 Metric results

The results of this experiment are significantly lower than the baseline (table 6.5), giving about 4% lower BLEU score in average. The produced text has almost no fluent sentences and the phenomena described above (missing words, wrong lexical choise etc.) were pretty obvious.

It seems that low performance is mainly a result of the incompatibility between the source and the target factors. As it can be seen in table 6.8, the arbitrary choice for

Table 6.8: Disproportion between English and Greek case tags

| case | English | Greek |
|---|---|---|
| nominative | 1 691 991 | 3 072 866 |
| accusative | 2 263 161 | 1 008 762 |
| genitive | 666 403 | 1 244 700 |
| vocative | 0 | 64 075 |
| no | 6 991 975 | 5 196 381 |

reducing the factors may not have been a good one, or at least did not comply with the way tagging had been performed in the English side. It is clear that the distribution of the "cases" in English is pretty improportional to the one in Greek. Meanwhile, even genitive, which was not ambiguous at all, appears to be quite improportional as well.

While there were plenty of ways to improve the data, which seem to be responsible for the obstacles of this experiment, it was not possible. Due to computational restrictions and the fact that the Greek data were finally available much more later than it had been planned, the possible parametrisations which would adhere to better results, still remain a challenge.

## 6.4 Using factors for verbs

This set of experiments deal with adding input factors, so as to determine the person of every verb. There is a hope that by helping the statistical system to disambiguate among disting target word forms, the translation accuracy and readability is improved.

### 6.4.1 Using only input factors

At first, we experimented by adding person factors on the input side, for every verb. Factors are obtained by identifying the subject of every sentence (section 5.3). Due to the fact that in English there is a distinct word form only for the third person, third person verbs were excluded from the factorisation to avoid sparse data; the factorisation is so far quite experimental and it often fails to give any tags for many of the verbs.

Table 6.9: BLEU and NIST score for experiments on verbs

BLEU

| | dev2006 | | devtest2006 | | test2007 | | avg |
|---|---|---|---|---|---|---|---|
| Baseline | 17.90 | | 18.05 | | 18.13 | | 18.09 |
| word+person→word | 18.08 | ∼ 88% | 18.05 | ∼ 50% | 18.06 | < 74% | 18.06 * |
| w+POS+person→w | 17.87 | ∼ 52% | 18.14 | ∼ 66% | 18.16 | ∼ 57% | 18.15 |
| w→w, per→per * | ∼ 1 − 2 | < 99% | ∼ 1 − 2 | < 95% | ∼ 1 − 2 | < 99% | ∼ 1 − 2 |

Due to hardware problems the experiment was not finished in time, after running for two weeks. Instead, we provide the estimate BLEU score, given after 9 runs of MERT (tuning)

NIST

| | dev2006 | | devtest2006 | | test2007 | | avg |
|---|---|---|---|---|---|---|---|
| Baseline | 5.216 | | 5.218 | | 5.279 | | 5.249 |
| word+person→person | 5.242 | ∼ 89% | 5.224 | ∼ 56% | 5.290 | ∼ 64% | 5.257 |
| word+POS+person→word | 5.232 | ∼ 78% | 5.259 | > 97% | 5.316 | > 95% | 5.288 |
| w→w, person→person | ∼ 1 | < 99% | ∼ 1 | < 99% | ∼ 1 | < 99% | ∼ 1 |

#### 6.4.1.1 Metric results

The results given by the metrics are not statistically significant according to the pairwise bootstrap verification. It is also possible to note that the NIST metric gives a better score. This may be explained by the fact that NIST scores long *n*-grams better: The syntax-based approach we have used for the factorization is much more efficient for resolving long-distance relationships, while short referrals could be easily handled by the baseline *n*-gram and the phrase-based probabilities anyway.

In a second experiment we combined English POS tags and person tags, motivated by the good performance of the POS experiments presented earlier. Indeed, the significant NIST metric results show improvement over the baseline, but no comparison can be done with the plain POS-tags experiment.

#### 6.4.1.2 Error analysis

Once again, introducing tags over the verbs has increased the number of the missing content words (table 6.10). Unknown words and wrong lexical choices seem to have increased as well. However, verb tags have managed to reduce the verb conjugation errors from 19% to 9%. On the one side, this indicates that the enrichment is in a good direction, but on the other side there still needs to be an effort to further reduce that 9%

of error, which may be due to parsing errors or missing factorisation rules.

It is also interesting that reducing the errors related to persons, increased the percentage of the errors related to other grammatical phenomena on verbs, i.e. mood, tense and voice. It was obvious through the results, that many sentences whose verbs had been corrected over the baseline, had now a correct person, but failed to capture other aspects of the verb choice. It is for sure, that since a backoff model was not possible to be built due to software restrictions, adding factors for mood, voice and tenses would definitely lead to sparse data, since they have very few distinct word forms and are not very frequent as phenomena.

At this point, the general impression is that when the verb conjugation has succeeded, the readability and adequacy of the text have improved, since it is now more clear what the constituents of the sentence are.

Table 6.10: Error analysis for experiment with factors on English verbs

| | baseline | w.+per. |
|---|---|---|
| Sense, reord. & lexical choice | | |
| Missing content words | 8.9% | 11.0% |
| Missing filler words | 10.8% | 10.0% |
| Local range order | 4.4% | 2.0% |
| Long range order | 4.4% | 2.0% |
| Wrong lexical choice | 15.7% | 17.1% |
| **Word form errors** | | |
| verb person | 18.9% | 9.0% |
| gender | 8.5% | 9.5% |
| pos | 4.2% | 3.8% |
| noun case VP | 4.4% | 5.1% |
| noun case PP | 5.7% | 11.0% |
| mood | 0.6% | 2.8% |
| tense | 0.6% | 1.5% |
| voice | 0.8% | 2.6% |
| **Various errors** | | |
| extra words | 8.7% | 6.1% |
| unknown words | 1.3% | 4.6% |
| punctuation | 0.4% | 0.3% |
| negative meaning | 1.5% | 1.5% |

### 6.4.2 Factors on both sides

Here, we tried to combine factors on both sides. For the Greek tags we used a concatenation of both the person and the number, since this better maps the way the English tags had been prepared. Unfortunately, there was once again a great issue of incompatibility between the two sides of the factors (table 6.12). One the one side, the Greek lexicon-based morphology tagger totally lacked information for many of the verbs; some of them had only a number tag, but not a person one. On the other side, our factorisation system does not seem to have managed to resolve a subject for all verbs.

Table 6.12: Number of verb factors on both sides

| person | English | Greek |
|---|---|---|
| 1st sing | 234 965 | 266 253 |
| 2nd sing | 30 564 | 129 186 |
| 3rd sing | 570 261 | 1 144 874 |
| 1st plural | 245 256 | 168 054 |
| 2nd plural | - | 35 922 |
| 3rd plural | 387 769 | 543 105 |
| no | 10 143 034 | 8 299 390 |

## 6.5 Experiment on gender

Here, we are trying to deal by a piece of information which cannot be produced given the source sentence data; instead, this experimental approach is trying to take advantage of a factored model, in order to "train" a gender translation table from the target sentence (section 5.4). Source side factorisation produced factors on a set of nouns (that would infer the gender of a noun-phrase); these factors were directly mapped to the gender tags that were given to nouns, adjectives and articles by the Greek morphology tagger. For the generation step, the inverted probability , as explained in 6.3.2, was used. The exact decoding process is illustrated in figure 6.4.

Figure 6.4: Experiment on gender: how translation components are mapped

### 6.5.1 Results

Table 6.13: BLEU and NIST score for experiment on gender

BLEU

| | dev2006 | | devtest2006 | | test2007 | | avg |
|---|---|---|---|---|---|---|---|
| Baseline | 17.90 | | 18.05 | | 18.13 | | 18.09 |
| ref→gender | 14.00 | < 99% | 13.88 | < 99% | 14.04 | < 99% | 13.96 |

NIST

| | dev2006 | | devtest2006 | | test2007 | | avg |
|---|---|---|---|---|---|---|---|
| Baseline | 5.216 | | 5.218 | | 5.279 | | 5.249 |
| ref→gender | 4.683 | < 99% | 4.680 | < 99% | 4.737 | < 99% | 4.709 |

The results given by this experiments are quite low, measuring about 4% lower BLEU score (table 6.13). We must also point out that the number of the source factors used is quite high, approximately equal to the number of the distinct English nouns, which is about 2.8 million words. This may have lead to unwanted pruning of many mappings, due to the way the probability direction was chosen (similar to what is show in equation 6.1, page 51).

The produced sentences have a high number of missing words. For the few gender mapping that have succeeded, we have to point out that there were errors on the number (e.g. we got some adjectives translated with a correct gender, but in singular instead of plural). This indicates that the number should also be taken into consideration, in a future model.

## 6.6 Combining factors altogether

As a conclusion, the outcome of the previous experiments was combined, so as to investigate how the factors would co-operate altogether. In this effort, a single phrase translation table was trained, by using both cases and verb-person as input factors. A second experiment also added English POS tags in the model. Finally, we experimented by combining the single phrase translation table of the former experiment, with the referral-to-gender phrase translation table of the previous section.

Table 6.14: BLEU and NIST score when using more than one factors

BLEU

| | dev2006 | | devtest2006 | | test2007 | | avg |
|---|---|---|---|---|---|---|---|
| Baseline | 17.90 | | 18.05 | | 18.13 | | 18.09 |
| word+person+case | 17.97 | ∼ 66% | 18.08 | ∼ 50% | 18.24 | ∼ 70% | 18.16 |
| word+POS+person+case | 17.90 | ∼ 51% | 18.11 | ∼ 67% | 18.02 | ∼ 62% | 18.07 |
| w+person+case→w, ref→gnd | 7.80 | < 99% | 7.85 | < 99% | 7.64 | < 99% | 7.75 |

NIST

| | dev2006 | | devtest2006 | | test2007 | | avg |
|---|---|---|---|---|---|---|---|
| Baseline | 5.216 | | 5.218 | | 5.279 | | 5.249 |
| word+person+case | 5.275 | > 100% | 5.258 | > 97% | 5.340 | > 98% | 5.299 |
| w+POS+per+case→w | 5.235 | ∼ 79% | 5.238 | ∼ 81% | 5.274 | ∼ 72% | 5.256 |
| w+per+case→w, ref→gnd | 4.003 | < 99% | 4.012 | < 99% | 4.023 | < 99% | 4.018 |

### 6.6.1 Results

While no BLEU score seems to be statistically significant, the NIST scores for the first experiment are quite encouraging (table 6.14). It appears that this combination gives the best score so far, raising it from 5.249 to 5.299. Knowing that the described methods are especially beneficial for long word sequence, NIST method seems to indicate that long *n*-grams have been improved. We could welcome this small score improvement as an indication that what has been explained was slightly useful in terms of improving the output. We would hope than there could be better results under better circumstances (e.g. better factorisation, more accurate tools, decoding back-off etc.).

## 6.7 Human evaluation

It is obvious, so far, that the metrics did not help reaching accurate conclusions. Apart from the manual word-by-word error analysis performed by the author, it was decided to conduct a human evaluation system, similar to the ones done for WMT tasks. Fourteen annotators were asked to judge a total of 268 groups of sample translations, with each group presenting the same translation by 4 indicative systems: the baseline, the one using POS tags, the one on verbs and the one that combines both persons and cases. The judgement was based by giving a 1 to 5 score on two factors: adequacy (how much the translation retains the meaning of the original sentence) and fluency

(how good is the produced sentence, in terms of grammar and syntax).

Table 6.15: Manual evaluation of adequacy and fluency

| system | adequacy | fluency |
|---|---|---|
| baseline | 3.47 | 3.45 |
| POS | 3.56 | 3.43 |
| person | 3.52 | 3.42 |
| person+case | 3.59 | 3.36 |

The results are shown in table 6.15. It appears that it was mostly agreed that sentences produced by the combined *word+person+case* model have a better way to give the meaning of the original sentence. This may be explained by the fact that we have focused on verbs and nouns, which are tightly connected with the meaning. Therefore, using methods to better indicate the constituency between these contextually important words, seems to be useful. The other enriched models seem to giver better adequacy than the baseline model, as well.

On the other side, it seems that none of the produced systems managed give a better fluency than the baseline. This has to do with the previously reported issues of sparse data.

## 6.8 Conclusion

This chapter gave the details of all the experiments that were performed, including detailed description and results for each of them. Evaluation included both metrics and manual error analysis, show as to focus on the specific problems that are being dealt. Each one of the methods seemed to be successful in terms of improving what it had been designed for, but of clear conclusions cannot be drawn due to the fact that not many of the metrics gave significant results. Though, one of targets seems to have been accomplished, since manual annotation indicated that combining factors could improve the adequacy of the produced translation.

# Chapter 7

# Conclusions and further work

## 7.1 Conclusions

We have been investigating whether SMT performance can be improved by adding linguistic information on the input, focusing on English-Greek translation. We initially considered three methods for preprocessing the English text. These methods focused on three linguistic phenomena which produce common errors on the output. In particular, noun cases, verb persons and gender of adjectives are required attributes by the target language, but not directly inferred by the source. For each of these sub-problems, our algorithm used heuristic syntax-based rules on the statistically generated syntax tree of each sentence, in order to create the missing information, which was tagged in by creating word factors.

The enriched input was used to create a set of SMT models on the chosen language pair of the Europarl corpus, using either factors on both sides, or single-sided ones. These experiments included thirteen different combinations of using the produced information, so as to gradually investigate the impact of the additions. The models were evaluated by using BLEU and NIST metrics and a pairwise bootstrap significance test, but additionally a manual word to word error analysis was performed, along with a manual adequacy/fluency evaluation.

Very few of our metrics results were significantly comparable to the baseline system. In the best measurable case, using both tags of verb persons and noun cases, NIST gave an improvement of 0.05, showing that a slight performance increase is significant when measuring difficult $n$-gram matches. Manual word to word evaluation showed that adding the tags for cases and persons reduced the number of the errors for each of these specific problems, but increased the number of untranslated/missing

words, an obvious indication of sparse data. There were efforts to eliminate these, by using a back-off translation component, but it was not possible to have that tested, due to software limitations and time restrictions. Apart from the sparse data and the lack of back-off decoding, it was shown that the low improvement is a profound effect of syntax-parser errors and the incapability to manually create tree-based rules that would fully cover all linguistic phenomena.

Finally, annotators who were asked to judge sample translations, concluded that the model which combined both persons and cases improved the adequacy (meaning) of the produced translation, but deteriorated the fluency. We could use the positive results given by the manual annotation and the metrics as a hint that the methods presented are in a good direction and could, under certain improvements, better address the problem.

## 7.2 Further work

Several aspects of the project were found to need improvement, but this was not possible due to the strict timeplan. Since the main problem after introducing factors was sparse data, there are reasons to believe that a back-off decoding would improve some issues, so that enabling backoff capabilities in the decoder should be the next step.

While using a second translation table to learn the gender of attributes seemed to have a basis as an idea, it didn't produce adequate results. More experimentation of possible combinations of that information could possible improve quality. Finally, we could consider further improvements on our decoder, by rescoring phrase-pairs using the linguistic or contextual information from the source sentence (CARPUAT and WU, 2007).

Our syntax parser was prone to several parsing errors and consistently failed to parse long/complicated sentences. Despite this problem was briefly addressed by using an approximation, we feel that all software tools used as a basis (syntax parser, POS tagger) need to be reconsidered, so that we finally choose some that would perform well on the current system. Also, a better aligned parallel corpus (possibly augmented with the extra data contained in the latest Europarl version) and a more handy target language morphology/syntax tagger would improve our training.

Part of the findings is that manually creating syntax-based rules are too slow and cannot easily cover all possible grammatical phenomena. An alternative would try to enrich input by extracting information from Parallel Grammars, using an XLE parser (Butt et al., 2002). While acquiring such a grammar is not easy either, this kind of

information seems to be more robust and adaptable to many translation pairs.

In an effort to avoid the drawbacks of using rules, we could also go back to a machine learning approach, where our input annotation would be learnt from the alignment of the Greek morphemes. That would require a quality target side tagger, whose tags would be mapped to the source side words; the described traversal rules may stand as features in this process.

Finally, we have to mention that since most of our work has been focused on the English side, all methods can be adapted for testing translation performance when translating into other morphologically rich languages, which would possibly benefit from such enrichment.

# Appendix A

# Aggregated results

Table A.1: BLEU scores

| | **BLEU** | | | | | | |
|---|---|---|---|---|---|---|---|
| | **dev2006** | | **devtest2006** | | **test2007** | | avg |
| w+person+case–>w , ref->gnd | 7.80 | 99% | 7.85 | 99% | 7.64 | 99% | 7.75 |
| word->word , ref->gnd , gnd<-word | 14.00 | 99% | 13.88 | 99% | 14.04 | 99% | 13.96 |
| word->word, case->case, case<-w | 13.84 | 99% | 13.78 | 99% | 13.97 | 99% | 13.88 |
| w->w, case->case, w+case->w | 1.29 | 99% | 1.47 | 99% | 1.26 | 99% | 1.37 |
| word+person+case->word | 17.97 | 66% | 18.08 | 50% | 18.24 | 70% | 18.16 |
| word+pos->word (no reord) | 17.76 | 83% | 17.95 | 79% | 17.84 | 98% | 17.90 |
| word+pos+person+case->word | 17.90 | 51% | 18.11 | 67% | 18.02 | 62% | 18.07 |
| word+pos+person->word | 17.87 | 52% | 18.14 | 66% | 18.16 | 57% | 18.15 |
| word+case->word | 17.69 | 89% | 17.58 | 49% | 17.94 | 91% | 17.76 |
| word+person->word | 18.08 | 88% | 18.05 | 50% | 18.06 | 74% | 18.06 |
| word+pos | 18.06 | 88% | 18.27 | 95% | 18.19 | 61% | 18.23 |
| word->word , pos->word | 1.13 | 99% | 1.39 | 99% | 1.15 | 99% | |
| baseline (5gram, max 60 words) | 17.900 | | 18.050 | | 18.130 | | 18.09 |

The percentage in every second column shows the significance of each set, if compared to the baseline (using pairwise bootstrap test). The average does not include the dev2006 since it was used for MERT tuning.

Table A.3: NIST scores

| | **NIST** | | | | | | |
|---|---|---|---|---|---|---|---|
| | **dev2006** | | **devtest2006** | | **test2007** | | avg |
| w+person+case–>w , ref->gnd | 4.003 | 99% | 4.012 | 99% | 4.023 | 99% | 4.018 |
| word->word , ref->gnd , gnd<-word | 4.683 | 99% | 4.680 | 99% | 4.737 | 99% | 4.709 |
| word->word, case->case, case<-w<br>w->w, case->case, w+case->w | 4.697 | 99% | 4.679 | 99% | 4.732 | 99% | 4.706 |
| word+person+case->word | 5.275 | 99% | 5.258 | 97% | 5.340 | 98% | 5.299 |
| word+pos->word (no reord) | 5.235 | 80% | 5.242 | 90% | 5.281 | 54% | 5.262 |
| word+pos+person+case->word | 5.235 | 79% | 5.238 | 81% | 5.274 | 72% | 5.256 |
| word+pos+person->word | 5.232 | 78% | 5.259 | 97% | 5.316 | 95% | 5.288 |
| word+case->word | 5.205 | 67% | 5.160 | 99% | 5.250 | 93% | 5.205 |
| word+person->word | 5.242 | 89% | 5.224 | 56% | 5.290 | 64% | 5.257 |
| word+pos<br>word->word , pos->word | 5.245 | 95% | 5.271 | 99% | 5.311 | 95% | 5.291 |
| baseline (5gram, max 60 words) | 5.216 | | 5.218 | | 5.279 | | 5.249 |

The percentage in every second column shows the significance of each set, if compared to the baseline (using pairwise bootstrap test). The average does not include the dev2006 since it was used for MERT tuning.

Table A.5: Manual error analysis

| | baseline | POS | persons | cases |
|---|---|---|---|---|
| Sense, reord. & lex. choice | | | | |
| Missing content words | 8.9% | 7.8% | 11.0% | 13.8% |
| Missing filler words | 10.8% | 8.4% | 10.0% | 9.6% |
| Local range order | 4.4% | 1.7% | 2.0% | 4.6% |
| Long range order | 4.4% | 1.8% | 2.0% | 6.1% |
| Wrong lexical choice | 15.7% | 14.7% | 17.1% | 14.7% |
| **Word form errors** | | | | |
| verb person | 18.9% | 21.3% | 9.0% | 15.5% |
| gender | 8.5% | 11.8% | 9.5% | 8.0% |
| pos | 4.2% | 2.9% | 3.8% | 2.9% |
| noun case VP | 4.4% | 4.3% | 5.1% | 2.5% |
| noun case PP | 5.7% | 8.9% | 11.0% | 4.2% |
| mood | 0.6% | 3.2% | 2.8% | 1.9% |
| tense | 0.6% | 1.4% | 1.5% | 2.1% |
| voice | 0.8% | 1.4% | 2.6% | 2.3% |
| **Various errors** | | | | |
| extra words | 8.7% | 5.8% | 6.1% | 7.8% |
| unknown words | 1.3% | 2.3% | 4.6% | 2.7% |
| punctuation | 0.4% | 0.3% | 0.3% | 0.2% |
| negative meaning | 1.5% | 2.0% | 1.5% | 1.0% |

The percentage in every second column shows the significance of each set, if compared to the baseline (using pairwise bootstrap test). The average does not include the dev2006 since it was used for MERT tuning.

# Bibliography

Birch, A., Osborne, M., and Koehn, P. (2007). CCG Supertags in factored Statistical Machine Translation. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 9–16, Prague, Czech Republic. Association for Computational Linguistics.

Brill, E. (1992). A simple rule-based part of speech tagger. *Proceedings of the Third Conference on Applied Natural Language Processing*, pages 152–155.

Brown, P., Della Pietra, S., Della Pietra, V., Lafferty, J., and Mercer, R. (1992). Analysis, statistical transfer, and synthesis in machine translation. *Proceedings of the Fourth International Conference on Theoretical and Methodological Issues in Machine Translation*, pages 83–100.

Brown, P. F. (1993). Applying statistical methods to machine translation. In *HLT '93: Proceedings of the workshop on Human Language Technology*, pages 397–397, Morristown, NJ, USA. Association for Computational Linguistics.

Brown, P. F., Cocke, J., Pietra, S. A. D., Pietra, V. J. D., Jelinek, F., Lafferty, J. D., Mercer, R. L., and Roossin, P. S. (1990). A statistical approach to machine translation. *Comput. Linguist.*, 16(2):79–85.

Brown, P. F., Pietra, S. A. D., Pietra, V. J. D., and Mercer, R. L. (1991). Word-sense disambiguation using statistical methods. In *Proceedings of the 29th annual meeting on Association for Computational Linguistics*, pages 264–270, Morristown, NJ, USA. Association for Computational Linguistics.

Brown, P. F., Pietra, V. J. D., Pietra, S. A. D., and Mercer, R. L. (1993). The mathematics of statistical machine translation: parameter estimation. *Comput. Linguist.*, 19(2):263–311.

Butt, M., Dyvik, H., King, T., Masuichi, H., and Rohrer, C. (2002). The Parallel Grammar project. *International Conference On Computational Linguistics*, pages 1–7.

CARPUAT, M. and WU, D. (2007). Improving Statistical Machine Translation using Word Sense Disambiguation. In *Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL 2007)*, pages 61–72, Prague, Czech Republic.

Collins, M. (1997). Three generative, lexicalised models for statistical parsing. *Proceedings of the 35th conference on Association for Computational Linguistics*, pages 16–23.

Collins, M., Koehn, P., and Kučerová, I. (2005). Clause restructuring for statistical machine translation. In *ACL '05: Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 531–540, Morristown, NJ, USA. Association for Computational Linguistics.

Durgar El-Kahlout, i. and Oflazer, K. (2006). Initial explorations in english to turkish statistical machine translation. In *Proceedings on the Workshop on Statistical Machine Translation*, pages 7–14, New York City. Association for Computational Linguistics.

El Isbihani, A., Khadivi, S., Bender, O., and Ney, H. (2006). Morpho-syntactic arabic preprocessing for arabic to english statistical machine translation. In *Proceedings on the Workshop on Statistical Machine Translation*, pages 15–22, New York City. Association for Computational Linguistics.

Goldwater, S. and McClosky, D. (2005). Improving statistical MT through morphological analysis. In *HLT '05: Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 676–683, Morristown, NJ, USA. Association for Computational Linguistics.

Huang, L., Knight, K., and Joshi, A. (2006). Statistical syntax-directed translation with extended domain of locality. *Proc. AMTA*, pages 66–73.

Jelinek, F. (1997). *Statistical methods for speech recognition*. MIT Press, Cambridge, MA, USA.

Koehn, P. (2004). Statistical significance tests for machine translation evaluation.

Koehn, P. (2005). Europarl: A parallel corpus for statistical machine translation. *MT Summit*, 5.

Koehn, P. (2007). Data intensive linguistics, lecture slides. retrieved from http://www.inf.ed.ac.uk/teaching/courses/emnlp/slides/emnlp16.pdf, May 2007.

Koehn, P. and Hoang, H. (2007). Factored translation models. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 868–876.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A., and Herbst, E. (2007). Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic. Association for Computational Linguistics.

Koehn, P. and Knight, K. (2003). Feature-rich statistical translation of noun phrases. In *ACL '03: Proceedings of the 41st Annual Meeting on Association for Computational Linguistics*, pages 311–318, Morristown, NJ, USA. Association for Computational Linguistics.

Koehn, P., Och, F. J., and Marcu, D. (2003). Statistical phrase-based translation. In *NAACL '03: Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, pages 48–54, Morristown, NJ, USA. Association for Computational Linguistics.

Lee, Y. (2004). Morphological analysis for statistical machine translation. *NAACL Proceedings*.

Marcu, D. and Wong, W. (2002). A phrase-based, joint probability model for statistical machine translation. In *EMNLP '02: Proceedings of the ACL-02 conference on Empirical methods in natural language processing*, pages 133–139, Morristown, NJ, USA. Association for Computational Linguistics.

Minkov, E., Toutanova, K., and Suzuki, H. (2007). Generating complex morphology for machine translation. In *ACL 07: Proceedings of the 45th Annual Meeting of the Association of Computational linguistics*, pages 128–135, Prague, Czech Republic. Association for Computational Linguistics.

Mitamura, T., Nyberg, E., Torrejon, E., Svoboda, D., Brunner, A., and Baker, K. (2002). Pronominal anaphora resolution in the kantoo multilingual machine translation system. *Proc. of TMI 2002*, pages 115–124.

Niessen, S. and Ney, H. (2001). Toward hierarchical models for statistical machine translation of inflected languages. In *Proceedings of the workshop on Data-driven methods in machine translation*, pages 1–8, Morristown, NJ, USA. Association for Computational Linguistics.

Niessen, S. and Ney, H. (2004). Statistical machine translation with scarce resources using morpho-syntactic information. *Comput. Linguist.*, 30(2):181–204.

NIST (2002). Automatic evaluation of machine translation quality using n-gram co-occurrence statistics.

Nitta, Y. (1986). Idiosyncratic gap: a tough prolem to structure-bound machine translation. In *Proceedings of the 11th coference on Computational linguistics*, pages 107–111, Morristown, NJ, USA. Association for Computational Linguistics.

Och, F., Tillmann, C., and Ney, H. (1999). Improved alignment models for statistical machine translation. *Proc. of the Joint SIGDAT Conf. on Empirical Methods in Natural Language Processing and Very Large Corpora*, pages 20–28.

Och, F. J. and Ney, H. (2003). A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2001). BLEU: a method for automatic evaluation of machine translation. In *ACL '02: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 311–318, Morristown, NJ, USA. Association for Computational Linguistics.

Petasis, G., Karkaletsis, V., Farmakiotou, D., Samaritakis, G., Androutsopoulos, I., and Spyropoulos, C. (2001). A Greek Morphological Lexicon and its Exploitation by a Greek Controlled Language Checker. *Proceedings of the 8th Panhellenic Conference on Informatics*, pages 8–10.

Petasis, G., Karkaletsis, V., Paliouras, G., and Spyropoulos, C. (2003). Using the ellogon natural language engineering infrastructure. *Proceedings of the Workshop on Balkan Language Resurces and Tools, 1st Balkan Conference in Informatics (BCI 2003)*.

Petasis, G., Paliouras, G., Karkaletsis, V., Spyropoulos, C., and Androutsopoulos, I. (1999). Resolving Part-of-Speech Ambiguity in the Greek Language Using Learning Techniques. *Arxiv preprint cs.CL/9906019*.

Stolcke, A. (2002). SRILM-an extensible language modeling toolkit. *Proc. ICSLP*, 2:901–904.

Trujillo, A. (1999). *Translation Engines: Techniques for Machine Translation*. Springer.

Ueffing, N. and Ney, H. (2003). Using pos information for statistical machine translation into morphologically rich languages. In *EACL '03: Proceedings of the tenth conference on European chapter of the Association for Computational Linguistics*, pages 347–354, Morristown, NJ, USA. Association for Computational Linguistics.

Vilar, D., Xu, J., D'Haro, L. F., and Ney, H. (2006). Error Analysis of Machine Translation Output. In *Proceedings of the 5th Internation Conference on Language Resources and Evaluation (LREC'06)*, pages 697–702, Genoa, Italy.

Wang, C., Collins, M., and Koehn, P. (2007). Chinese syntactic reordering for statistical machine translation. pages 737–745.

Yamada, K. and Knight, K. (2001). A syntax-based statistical translation model. In *ACL '01: Proceedings of the 39th Annual Meeting on Association for Computational Linguistics*, pages 523–530, Morristown, NJ, USA. Association for Computational Linguistics.

Zhang, Y., Vogel, S., and Waibel, A. (2004). Interpreting BLEU/NIST scores: How much improvement do we need to have a better system. *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC)e*, pages 2051–2054.