An Divide-and-Conquer Strategy for Recognizing Textual Entailment

Rui Wang
Saarland University
Saarbruecken, Germany
rwang@coli.uni-sb.de

Guenter Neumann
DFKI GmbH
Saarbruecken, Germany
neumann@dfki.de

Abstract

This paper describes our participation of the Recognizing Textual Entailment challenge this year. Based on our promising results in the RTE-3 challenge last year (66.9% of accuracy) using a precision-oriented puristic syntactic approach (puristic in the sense that we only performed dependency parsing), we explored further extensions of this perspective. By extension, we developed more specialized RTE-modules to tackle more cases (i.e., entailment pairs) while trying to keep high accuracy.

1 Introduction

In (Wang and Neumann, 2007a) we developed a puristic syntactic approach to the problem of Recognizing Textual Entailment (RTE) by only performing a syntactic dependency analysis on corresponding text pairs. With 66.9% accuracy, we achieved quite promising results on the test data of the RTE-3 challenge, which ranked us among the 5th best systems, cf. (Giampiccolo et al., 2007). In order to test the application potential of our approach, we also utilized the system as a core engine and integrated it with other linguistic processing modules for the Answer Validation Exercise[1] at the Cross Language Evaluation Forum[2] (AVE@CLEF) and obtained the best result for English (Wang and Neumann, 2007b; Wang and Neumann, 2008) and for German (Wang and Neumann, 2008).

Our puristic approach actually gives us a high syntactic-oriented baseline. This is achieved by constructing structural features from the abstract tree descriptions, which are automatically extracted from syntactic dependency trees. These features are then applied by a subsequence-kernel-based classifier that learns to decide whether the entailment relation holds between two texts. Note that the classifier is actually only applied on those pairs <**T**,**H**>, for which the (binary) dependency tree of **H** can be identified as a subtree of **T**. In order to further explore this line of research, we are interesting in investigating new RTE-technology on top of this approach through the integration of lexical-semantic features, such that a finer-grained control on the subtree relationship can be captured.

In particularly, we have developed a new divide-and-conquer architecture for RTE. The core idea is to provide a set of specific RTE methods and to combine them with a voting mechanism. Each RTE-specialist considers only a specific kind of RTE problem, so that the expected accuracy can be maximized. In the current version, we have developed and implemented three RTE-specialists:

- Temporal anchored pairs

Extract temporal expressions and corresponding events from the dependency trees, and apply entailment rules between extracted time event pairs;

- Named entity pairs

Extract other Named Entities (NE) and corresponding events, and apply entailment rules between extracted entity-event pairs;

- Noun phrase anchored pairs

For pairs with no NEs but two consisting NPs, determine the subtree alignment, and apply a kernel-based classifier.

The latter case corresponds to the subsequence-kernel-based approach mentioned above, and has been enumerated here in order to demonstrate the parallel and independent status of the different RTE-specialists. In addition to

[1] http://nlp.uned.es/clef-qa/ave/
[2] http://www.clef-campaign.org/

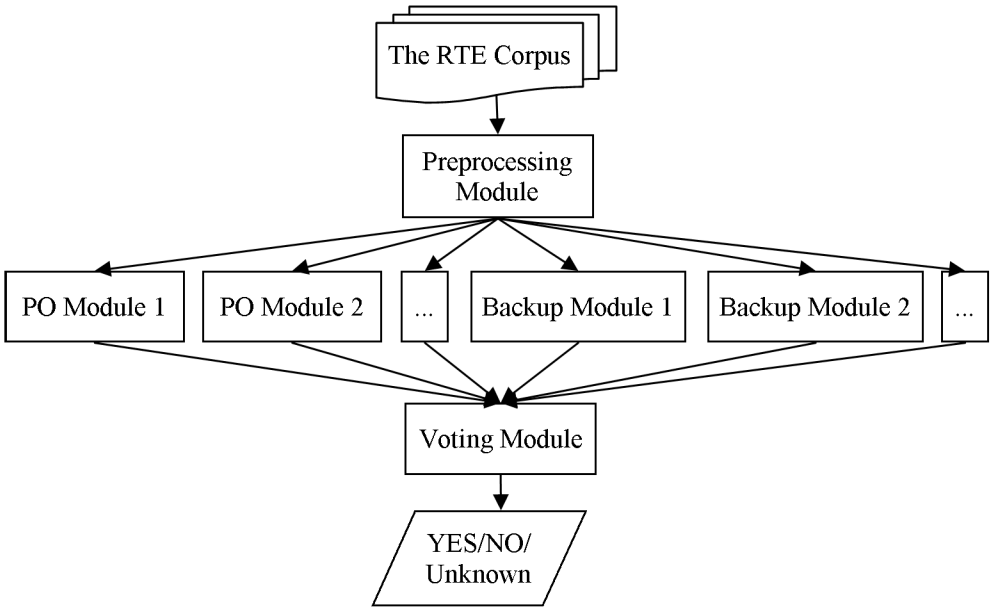

Figure 1 Architecture of the system

the accuracy-oriented RTE-modules, we also consider two robust but not necessarily precise backup strategies that deal with those cases which cannot be covered by any specialist.

In the rest of the paper, we will introduce the architecture of our system (Section 2), and describe in detail each specialized RTE-modules in Section 3. The experimental results will be shown in Section 4 and followed by a discussion in Section 5. The last section will conclude the paper.

2 Architecture

Figure 1 shows the architecture of the whole system, which contains a preprocessing module, a voting module and several Precision-Oriented (PO) RTE modules. For preprocessing, we utilize several linguistic processing components, such as a POS tagger, a dependency parser[3], and a named-entity (NE) recognizer[4] to annotate the original plain texts from the RTE corpus. We then apply several specialized PO RTE-modules, each of which is responsible for a subset of the whole data set. Since all the modules aim at high precision, they do not necessarily cover all the entailment pairs. The cases which cannot be covered by any specialized RTE-module will be passed to the high-coverage, but probably less accurate backup modules. In the final stage, we join the results of all specialized RTE-modules and backup modules together. By doing so, different confidence values are assigned to the different RTE-modules according to the performances on training data. In order to deal with possible overlapping cases (i.e. entailment pairs that are covered by more than one module), a voting mechanism is applied that takes into account the confidence values.

3 Specialized Modules and Backup Modules

Based on the architecture we proposed in the previous section, we can easily add or remove specialized RTE-modules and backup modules. In this paper, we consider three specialized PO RTE-modules and two backup modules.

3.1 The TACTE Module (TAC-M)

The TACTE (Time Anchoring Component for Textual Entailment) module was proposed by Wang and Zhang (2008). The basic process consists of three main steps: 1) extracting and anchoring temporal expressions; 2) using temporal expressions as starting points to and corresponding events in the dependency structure; and 3) applying lexical resources and entailment rules between temporal expressions to detect the entailment relationship.

For example, we have a *Text-Hypothesis* (**T-H**) pair as follows,

T: **Released** in **1995**, *Tyson returned to boxing,* **winning** the World Boxing Council title in **1996**. The same year,

[3] We used Minipar (Lin, 1998) for both POS tagging and dependency parsing.

[4] We used the Stanford NE recognition system (Finkel et al., 2005).

however, he lost to Evander Holyfield, and in a **1997** **rematch** **bit** Holyfield's ear, for which he was temporarily banned from boxing.
H: In **1996** Mike Tyson ***bit*** Holyfield's ear.

In the first step, we extract temporal expressions, e.g. *1995*, *1996*, and *1997*. The TAC (Time Anchoring Component) can deal with both absolute temporal expressions and relative temporal expressions. In the above example, these are all the former ones. Examples for the latter case will be *yesterday*, *the next year*, etc. Provided with a reference date (given by the context or predefined manually), the relative temporal expressions can be normalized into corresponding absolute ones.

There are two other issues concerning the temporal expressions, the granularity and the types. We use the following granularity order to normalize all the temporal expressions, and we also simplify all the temporal expressions into two categories, the time point (*on 6th of May, 1983*) and the interval (*from Wednesday to Saturday*).

second < minute < hour < pofd < dofw < day < weeknumber < pofm < month < pofy < year[5]

In the second step, we locate the temporal expressions in the dependency tree and then traverse the nodes on the tree to find the nearest verb or noun[6]. Since in most cases, the temporal expression is either a modifier of a noun phrase or a part of a verb phrase modifier (usually the latter is realized as a prepositional phrase). The goal of this procedure is to find the corresponding nouns or verbs which the temporal expressions modify. The dependency structure of this example is shown partially in Figure 2,

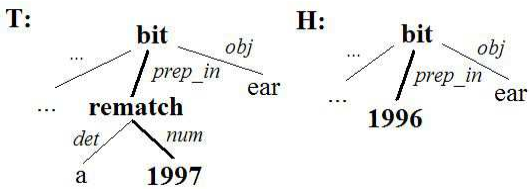

Figure 2 Partial dependency trees of the example

After applying the second step, the following events could be extracted,

T: *1995*: *released* (verb); *1996*: *winning* (nominalization); *1997*: *rematch* (noun), *bit* (verb)
H: *1996*: *bit* (verb)

In the last step, we derive a new feature representation from the input textual pairs. Instead of computing the surface string similarity, we now compare two pairs of temporal expressions and their corresponding events. Such pairs are defined as *EventTimePairs* (ETPs), and each of them consists of a noun or a verb denoting the event and the corresponding temporal expression. In order to resolve the relation between two ETPs, we need to separately resolve the relation between events and between temporal expressions, and combine the results afterwards. For the former, we make use of lexical resources, e.g. WordNet (Miller et al., 1993), VerbOcean (Chklovski and Pantel, 2004) to discover the relationship between two events (i.e. nouns or verbs); and for the latter, we manually define entailment rules shown in the following table[7],

	P → P	P → D	D → P	D → D
Same	IDENTITY	NO	INCLUDE	INCLUDE
F → C	INCLUDED	NO	INCLUDE	INCLUDE
C → F	NO	IDENTITY	INCLUDE	INCLUDE

Table 1 Entailment rules between temporal expressions

Finally, we combine the results together. Either the entailment between temporal expressions or between events does not hold, the entailment between the two ETPs does not hold; otherwise, in principle, it is unknown, since other information might negate the answer. Consequently, only if all the ETPs in **H** cannot be entailed by the ETPs in **T**, the final answer to that **T-H** pair is *NO*; otherwise, it is unknown. The results for the above example are,

- <*release*, *1995*>, <*bit*, *1996*>: *NO*
- <*win*[8], *1996*>, <*bit*, *1996*>: *NO*
- <*rematch*, *1997*>, <*bit*, *1996*>: *NO*
- <*bit*, *1997*>, <*bit*, *1996*>: *NO*

Therefore, the final answer is *NO*. Notice that TAC-M can be only applied on those pairs,

[5] pofd: part-of-day, dofw: day-of-week, pofm: part-of-month, pofy: part-of-year.
[6] In this paper, we assume that an *event* can be either represented by a noun (including nominalizations) or by a verb.
[7] P refers to time points, D refers to duration, F and C refer to fine and coarse granularity respectively. NO means no entailment; otherwise, the entailment holds.
[8] After applying lexical resources to change the nominalization back into the original verb form.

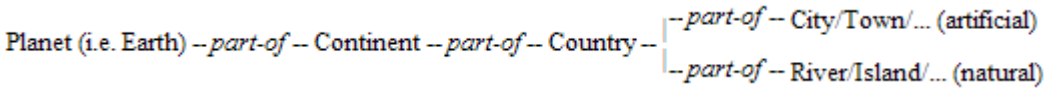

Figure 3 The basic structure of the geographic ontology

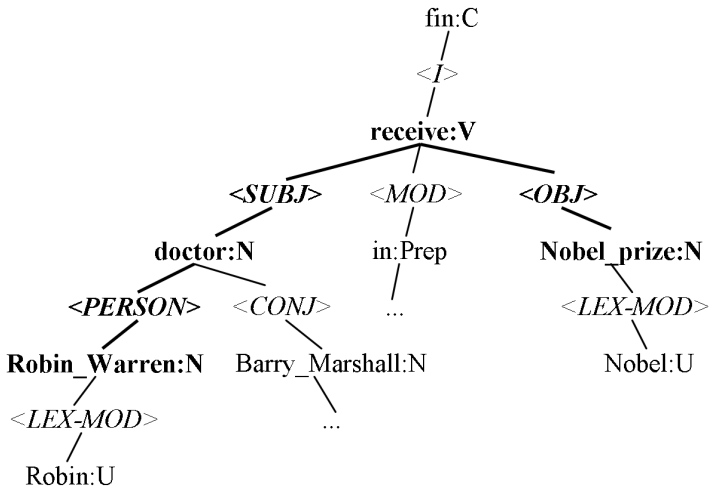

Figure 4 An example of the tree skeleton (in bold)

which both **T** and **H** contain temporal expressions.

3.2 The NE-Oriented Module (NE-M)

We also extended the approach described above to other NE types, i.e. person *names* (PNs), location *names* (LNs), and *organization names* (ONs). The process is quite similar to TAC-M, replacing the temporal expressions by other NEs. Therefore, the ETP can be extended into the following Event structure,

<Event, Time, Location, List<**Participants**>>

Event can be either a noun or a verb; *Time* is a normalized temporal expression; *Location* is an LN; a *Participant* can be either a PN or an ON.

In particular, after referring several geographic taxonomies (Geonames[9], WorldGazetteer[10], etc.), we construct a geographic ontology using geographic terms and two relations. The backbone taxonomy of the ontology is shown in Figure 3. The structure consists of geographic terms referring different granularities of areas. Inside each Country, we have two categories of fine-grained places, i.e. artificial divisions and natural places. The basic relation in-between is the directional part-of relation, which means the geographic area on the left side contains the area on the right side.

In addition, extra geographic areas are connected with these basic terms using the same part-of relation. For example, the following geographic areas consist of the basic terms above,

Subcontinent: *the Indian subcontinent*, *the Persian Gulf*, etc.
Subcountry: *Lower Saxony*, *the Western USA*, etc.

An additional equal relation is utilized for synonyms and abbreviations of the same geographic area, e.g. *the United Kingdom*, *the UK*, *Great Britain*, etc.

Consequently, the entailment rules between Events also have more dimensions. However, in one word, all the information contained in **H** must be fully entailed by **T**; otherwise, it is *NO*.

3.3 The Tree Skeleton Module (TS-M)

This module implements the main approach proposed by Wang and Neumann (2007a). The main idea is to extract a new sentence representation called *Tree Skeleton* (TS) based on the dependency parse trees, and then use a kernel-based machine learning method to make the prediction of the entailment relation.

The TS structure can be viewed as an extended version of the predicate-argument structure. Since it contains not only the predicate and its arguments, but also the dependency paths in-between, it captures the essential part of the sentence. Following their algorithm, we first select overlapping topic words (i.e. nouns) in **T**

[9] Geonames geo coding web service: http://www.geonames.org/
[10] WorldGazetteer: http://www.world-gazetteer.com

and **H** (we use fuzzy match at the substring level instead of full match). Starting with these nouns, we traverse the dependency tree to identify the lowest common ancestor node (named as *root node*). This sub-tree without the inner yield is defined as a *Tree Skeleton*. Figure 3 shows the TS of **T** in the following **T**-**H** pair,

T: For their discovery of ulcer-causing bacteria, Australian doctors **Robin Warren** and Barry Marshall have ***received*** the 2005 **Nobel Prize** in Physiology or Medicine.
H: **Robin Warren** was **awarded** *a* **Nobel Prize**.

The current version of the TS only deals with **T**-**H** pairs which contain two dependency paths. In experiments, tree skeletons can be successfully extracted from more than 30% of the previous RTE data sets.

The extracted TSs of **T** and **H** for this example will be as follows,

T: Robin_Warren:N <PERSON> doctor:N <SUBJ> receive:V <OBJ> Nobel_prize:N
H: Robin_Warren:N <OBJ1> award:V <OBJ2> Nobel_prize:N

After generalization of the dependency labels and also deletion of the word forms, we utilized subsequence kernel (Bunescu and Mooney, 2005) to represent the differences between the two TSs. Please refer Wang and Neumann (2007a) for more details.

3.4 The Backup Modules

Chief requirements for the backup strategy are robustness and simplicity. Therefore, we considered two backup modules, the Triple backup (Tri-BM) and the Bag-of-Words (BoW) backup (BoW-BM) (Wang and Neumann, 2007a). The former one is based on the Triple similarity function which operates on two triple (dependency structure represented in the form of <*head, relation, modifier*>) sets and determines how many triples of **H** are contained in **T**. The core assumption here is that the higher the number of matching triple elements, the more similar both sets are, and the more likely it is that **T** entails **H**. The function uses an approximate matching function. Different cases (i.e. ignoring either the parent node or the child node, or the relation between nodes) might provide different indications for the similarity of **T** and **H**. We then sum them up using different weights and divide the result by the cardinality of **H** for normalization.

The BoW-BM is based on BoW similarity score, which is calculated by dividing the number of overlapping words between **T** and **H** by the total number of words in **H** after a simple tokenization according to the space between words.

4 Experimental Results

For the participation of the challenge, we submitted three runs for each TAC RTE task, which differ in assignment of different confidence values to the used RTE-modules. The configurations of the three submissions for the two-way task and the results are as follows,

- Run1: TAC-M, TS-M, and Tri-BM
- Run2: TAC-M, TS-M, and BoW-BM
- Run3: TAC-M, TS-M, NE-M, and Tri-BM, BoW-BM

According to the performances of the modules on the development sets, the voting model is simply taking the results from the module which has highest accuracy. Those pairs, which are not covered by any specialized modules, will be passed to backup modules, since they always have answers.

Compared to our best result in RTE-3 challenge, there is an improvement of 3.7% of accuracy. In particular, the TAC-M has the highest accuracy, though the coverage is the lowest. The performance of TS-M is higher than the average accuracy, showing the advantage of the tree skeleton structure. NE-M does not have a good accuracy, which is contradictive to what we aimed at. The lower performance of the NE recognition (compared with temporal expressions) might be a cause.

The configurations of the three submissions for the three-way task and the results are,

- Run1: TAC-M, TS-M, and Tri-BM, BoW-BM
- Run2: TAC-M, TS-M, NE-M (partial), and Tri-BM, BoW-BM
- Run3: TAC-M, TS-M, NE-M, and Tri-BM, BoW-BM

Since our modules were not specially designed for recognizing three-way entailment, we take a strategy to combine results from different modules. For specialized modules, we keep *YES* as *ENTAILMENT*, but change *NO* into

Tasks	TAC-M	TS-M	NE-M	BoW-BM	Tri-BM	Run1	Run2	Run3
IR(300)	75.0%/4	**76.5%/85**	61.0%/164	63.3%	54.3%	66.0%	**72.3%**	71.7%
QA(200)	**90.0%/10**	73.2%/82	54.8%/93	49.0%	53.5%	73.0%	72.0%	**74.0%**
SUM(200)	**83.3%/6**	74.5%/51	55.2%/67	63.5%	54.0%	64.0%	69.5%	**71.5%**
IE(300)	72.7%/11	**74.2%/128**	46.7%/152	50.0%	50.0%	**66.7%**	66.3%	**66.7%**
All(1000)	**80.6%/31**	74.6%/346	54.3%/477	56.5%	52.8%	67.2%	69.9%	**70.6%**

Table 2 Results of all modules and submissions for the two-way RTE task

Answers	Run1(2)	Run2(2)	Run3(2)	Answers	Run1(3)	Run2(3)	Run3(3)
Yes(500)	66.6%	**81.4%**	74.8%	**Yes(500)**	68.2%	66.6%	**72.8%**
No(500)	**67.8%**	58.4%	66.4%	**No(150)**	38.7%	**41.3%**	33.3%
/	/	/	/	**Unknown(350)**	**61.4%**	47.1%	54.9%
All(1000)	67.2%	69.9%	**70.6%**	**All(1000)**	**61.4%**	56.0%	60.6%

Table 3 Results of all submissions for both the two-way and three-way RTE tasks

UNKNOWN. For the backup modules, we take the following rules,

- If BoW-BM=*YES* & Tri-BM=*NO* then *CONTRADICTION*
- If BoW-BM=*YES* & Tri-BM=*YES* then *ENTAILMENT*
- Others *UNKNOWN*

Comparing the two-way task and the three-way task, we find that *CONTRADICTION* cases are not trivial to capture (only around 40% of accuracy), whose difficulty and importance are also discussed by de Marneffe et al. (2008).

To sum up, not only the results are quite satisfactory, but also we obtain good indicators for deciding which entailment cases can be more reliably handled by which RTE-module. The latter is a very promising direction to be further explored in the future.

5 Discussion

Though we have not done a detailed error analysis yet, the preliminary observations already show some interesting issues. Our results on the TAC-M and NE-M modules are consistent with Herrera et al. (2005) and Vanderwende et al. (2006), which showed the effectiveness of NE features. They actually encode the NE information as feature values, which makes it difficult to check the explicit contribution of NE information. Our approach is much more transparent to this, because we explicitly select a NE subset for which we can demonstrate its benefits.

Bobrow et al. (2007) also propose a precision-oriented approach, however we a much lower coverage on the whole data set. MacCartney amd Manning (2007) applied natural logic to the RTE task, and also dealt with specific cases of entailment pairs, e.g. quantifiers. Many other approaches also explore the limitation of coverage, e.g. using lexical-syntactic rules (Bar-Haim et al., 2007). It seems the RTE task cannot easily be solved by using only a single generic method (ako generic problem solver), but might benefit from the the combination of different approaches.

In fact, many researchers have been focusing on the integration of different approaches. Bos and Markert (2005) combined a rigid logic inference system with shallow lexical features to gain from both sides. MacCartney amd Manning (2007) also applied a shallow system in order to achieve the full coverage of the data set, which is similar to our backup modules. Our particular contribution is the ranking of different modules based on their confidence values, so that a high precision could be maximally preserved.

6 Conclusion

In this paper, we have described our system for this year's RTE challenge at TAC 2008. The main idea is to advocate a divide-and-conquer strategy to utilize specialized RTE-modules to deal with specific entailment cases. The key requirement for these modules is high precision, while the coverage need not necessarily be high. In order to combine all modules' results, we rank the modules on basis of confidence values that have been automatically derived from a

performance analysis using training data. Our result is quite consistent with other researcher's work, and it seems to indicate an effective way of handling this challenging task.

Acknowledgments
The work presented here was partially supported by the EC-funded project QALL-ME and Rui Wang is funded by the PIRE PhD scholarship program.

References

Bar-Haim, R., Dagan, I., Greental, I., Szpektor, I., and Friedman, M. 2007. Semantic inference at the lexical-syntactic level for textual entailment recognition. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pages 131–136, Prague.

Bobrow, D., Crouch, D., King, T., Condoravdi, C., Karttunen, L., Nairn, R., de Paiva, V., and Zaenen, A. 2007. Precision-focused Textual Inference. In *Proceedings of the ACL Workshop on Textual Entailment and Paraphrasing*, Prague.

Bos, J. and Markert, K. 2005. Combining Shallow and Deep NLP Methods for Recognizing Textual Entailment. In *Proceedings of PASCAL Workshop on Recognizing Textual Entailment*, Southampton, UK.

Bunescu, R and Mooney, R. 2005. Subsequence kernels for relation extraction. In *Proceedings of the 19th Conference on Neural Information Processing Systems*, Vancouver, British Columbia.

Chklovski, T. and Pantel, P. 2004. Verbocean: Mining the web for fine-grained semantic verb relations. In *Proceedings of Conference on Empirical Methods in Natural Language Processing (EMNLP-04)*, volume 34, pages 343-360, Barcelona, Spain.

de Marneffe, M., Rafferty A., and Manning, C. 2008. Finding contradictions in text. In *Proceedings of ACL-HLT 2008*.

Finkel, J., Grenager, T., and Manning, C. 2005. Incorporating Non-local Information into Information Extraction Systems by Gibbs Sampling. In Proceedings of the 43nd Annual Meeting of the Association for Computational Linguistics (ACL 2005), pp. 363-370.

Giampiccolo, D., Magnini, B., Dagan, I., and Dolan, B. 2007. The Third PASCAL Recognizing Textual Entailment Challenge. In *Proceedings of the ACL Workshop on Textual Entailment and Paraphrasing*, Prague.

Herrera, J., Peñas. A., and Verdejo, F. 2005. Textual Entailment Recognition Based on Dependency Analysis and WordNet In *Proceedings of PASCAL Workshop on Recognizing Textual Entailment*, Southampton, UK.

Lin, D. 1998. Dependency-based evaluation of MINIPAR. In *Proceedings of the Workshop on the Evaluation of Parsing Systems*, Granada, Spain.

MacCartney, B. and Manning, C. 2007. Natural Logic for Textual Inference. In *Proceedings of the ACL Workshop on Textual Entailment and Paraphrasing*, Prague.

Miller, G., Beckwith, R., Fellbaum, C., Gross, D., and Miller, K. 1993. Five papers on WordNet. Technical report, Cognitive Science Laboratory, Princeton University.

Vanderwende, L., Menezes, A., and Snow, R. 2006. Microsoft Research at RTE-2: Syntactic Contributions in the Entailment Task: an implementation. In *Proceedings of the Second PASCAL Challenges Workshop on Recognising Textual Entailment*, Venice, Italy.

Wang, R. and Neumann, G. 2007a. Recognizing Textual Entailment Using a Subsequence Kernel Method. In *Proceedings of the Twenty-Second AAAI Conference on Artificial Intelligence (AAAI-07)*, July 22-26, Vancouver, Canada.

Wang, R. And Neumann, G. 2007b. DFKI–LT at AVE 2007: Using Recognizing Textual Entailment for Answer Validation. In *online proceedings of CLEF 2007 Working Notes*, ISBN: 2-912335-31-0, September 2007, Budapest, Hungary.

Wang, R. and Neumann, G. 2008. Information Synthesis for Answer Validation. In *Working Notes for the CLEF 2008 Workshop*, 17-19 September, Aarhus, Denmark.

Wang, R. and Zhang, Y. 2008. Recognizing Textual Entailment with Temporal Expressions in Natural Language Texts. In *Proceedings of the IWSCA-2008*, Pages 109-116, IEEE Computer Society.