Recognizing Textual Relatedness with Predicate-Argument Structures

Rui Wang
Dept of Computational Linguistics
Saarland University
66123 Saarbrücken, Germany
rwang@coli.uni-sb.de

Yi Zhang
Dept of Computational Linguistics
Saarland University
LT-Lab, DFKI GmbH
D-66123 Saarbrücken, Germany
yzhang@coli.uni-sb.de

Abstract

In this paper, we first compare several strategies to handle the newly proposed three-way *Recognizing Textual Entailment* (RTE) task. Then we define a new measurement for a pair of texts, called *Textual Relatedness*, which is a weaker concept than semantic similarity or paraphrase. We show that an alignment model based on the predicate-argument structures using this measurement can help an RTE system to recognize the *Unknown* cases at the first stage, and contribute to the improvement of the overall performance in the RTE task. In addition, several heterogeneous lexical resources are tested, and different contributions from them are observed.

1 Introduction

Recognizing Textual Entailment (RTE) (Dagan et al., 2006) is a task to detect whether one *Hypothesis* (H) can be inferred (or entailed) by a *Text* (T). Being a challenging task, it has been shown that it is helpful to applications like question answering (Harabagiu and Hickl, 2006). The recent research on RTE extends the two-way annotation into three-way[1] [2], making it even more difficult, but more linguistic-motivated.

The straightforward strategy is to treat it as a three-way classification task, but the performance suffers a significant drop even when using the same classifier and the same feature model. In fact, it can also be dealt with as an extension to the traditional two-way classification, e.g., by identifying the *Entailment* (E) cases first and then further label the *Contradiction* (C) and *Unknown* (U) T-H pairs. Some other researchers also work on detecting negative cases, i.e. contradiction, instead of entailment (de Marneffe et al., 2008). However, according to our best knowledge, the detailed comparison between these strategies has not been fully explored, let alone the impact of the linguistic motivation behind the strategy selection. This paper will address this issue.

Take the following example from the RTE-4 test set (Giampiccolo et al., 2009) as an example,

T: *At least five people have been killed in a head-on train collision in north-eastern France, while others are still trapped in the wreckage. All the victims are adults.*

H: *A French train crash killed children.*

This is a pair of two contradicting texts, the mentioning of events (i.e. train crash) in both T and H are assumed to refer the same event[3]. In fact, the only contradicting part lies in the second sentence of T against H, that is, whether there are children among the victims. Therefore, this pair could also be classified as a *Known* (K) pair (=E∪C) against *Unknown* (U) pairs, instead of being classified as a *Non-entailment* (N) case (=C∪U) against E case in the traditional two-way annotation.

Furthermore, many state-of-the-art RTE approaches which are based on overlapping information or similarity functions between T and H, in fact over-cover the E cases, and sometimes, cover the C cases as well. Therefore, in this paper, we

[1] http://nlp.stanford.edu/RTE3-pilot/

[2] http://www.nist.gov/tac/tracks/2008/rte/rte.08.guidelines.html

[3] See more details about the annotation guideline at http://www.nist.gov/tac/tracks/2008/rte/rte.08.guidelines.html

Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, pages 784–792,
Singapore, 6-7 August 2009. ©2009 ACL and AFNLP

would like to test whether applying this style of approaches to capture the K cases instead of E cases is more effective. While in lexical semantics, semantic relatedness is a weaker concept than semantic similarity, there is no counterpart at the sentence or text level. Therefore, in this paper, we propose a *Recognizing Textual Relatedness (RTR)* task as a subtask or the first step of RTE. By doing so, we choose *predicate-argument structure (PAS)* as the feature representation, which has already been shown quite useful in the previous RTE challenges (Wang and Neumann, 2007).

In order to obtain the PAS, we utilize a Semantic Role Labeling (SRL) system developed by Zhang et al. (2008). Although SRL has been shown to be effective for many tasks, e.g. information extraction, question answering, etc., it has not been successfully used for RTE, mainly due to the low coverage of the verb frame or semantic role resources or the low performance of the automatic SRL systems. The recent CoNLL shared tasks (Surdeanu et al., 2008; Hajič et al., 2009) have been focusing on semantic dependency parsing along with the traditional syntactic dependency parsing. The PAS from the system output is almost ready for use to build applications based on it. Therefore, another focus of this paper will be to apply SRL to the RTE task. In particular, it can improve the first stage binary classification (K vs. U), and the final result improves as well.

The rest of the paper will be organized as follows: Section 2 will give a brief literature review on both RTE and SRL; Section 3 describes the semantic parsing system, which includes a syntactic dependency parser and an SRL system; Section 4 presents an algorithm to align two PASs to recognize textual relatedness between T and H, using several lexical resources; The experiments will be described in Section 5, followed by discussions; and the final section will conclude the paper and point out directions to work on in the future.

2 Related Work

Although the term of *Textual Relatedness* has not been widely used by the community (as far as we know), many researchers have already incorporated modules to tackle it, which are usually implemented as an alignment module before the inference/learning module is applied. For example, Pado et al. (2009) mentioned two alignment modules, one is a phrase-based alignment system called MANLI (MacCartney et al., 2008), and the other is a stochastic aligner based on dependency graphs. Siblini and Kosseim (2009) performed the alignment on top of two ontologies. In this paper, we would like to follow this line of research but on another level of representation, i.e. the predicate-argument structures (PAS), together with different lexical semantic resources.

As for the whole RTE task, many people directly do the three-way classification with selective features (e.g. Agichtein et al. (2009)) or different inference rules to identify entailment and contradiction simultaneously (e.g. Clark and Harrison (2009)); while other researchers also extend their two-way classification system into three-way by performing a second-stage classification afterwards. An interesting task proposed by de Marneffe et al. (2008) suggested an alternative way to deal with the three-way classification, that is, to split out the contradiction cases first. However, it has been shown to be more difficult than the entailment recognition. Based on these previous works and our experimental observations, we propose an alternative two-stage binary classification approach, i.e. to identify the unknown cases from the known cases (entailment and contradiction) first. And the results show that due to the nature of these approaches based on overlapping information or similarity between T and H, this way of splitting is more reasonable.

However, RTE systems using semantic role labelers has not shown very promising results, although SRL has been successfully used in many other NLP tasks, e.g. information extraction, question answering, etc. According to our analysis of the data, there are mainly three reasons: a) the limited coverage of the verb frames or predicates; b) the undetermined relationships between two frames or predicates; and c) the unsatisfying performance of an automatic SRL system. For instance, Burchardt et al. (2007) attempted to use FrameNet (Baker et al., 1998) for the RTE-3 challenge, but did not show substantial improvement. With the recent CoNLL challenges, more and more robust and accurate SRL systems are ready for use, especially for the PAS identification. For the lexical semantics, we also discover that, if we relax the matching criteria (from similarity to relatedness), heterougeous resources can contribute to the coverage differently and then the effectiveness of PAS will be shown as well.

3 Semantic Parsing

In order to obtain the predicate-argument structures for the textual entailment corpus, we use the semantic role labeler described in (Zhang et al., 2008). The SRL system is trained on the Wall Street Journal sections of the Penn Treebank using PropBank and NomBank annotation of verbal and nominal predicates, and relations to their arguments, and produces as outputs the semantic dependencies. The head words of the arguments (including modifiers) are annotated as a direct dependent of the corresponding predicate words, labeled with the type of the semantic relation (Arg0, Arg1 . . . , and various ArgMs). Note that for the application of SRL in RTE task, the PropBank and NomBank notation appears to be more accessible and robust than the the FrameNet notation (with much more detailed roles or frame elements bond to specific verb frames).

As input, the SRL system requires syntactic dependency analysis. We use the open source MST Parser (McDonald et al., 2005), trained also on the Wall Street Journal Sections of the Penn Treebank, using a projective decoder with second-order features. Then the SRL system goes through a pipeline of 4-stage processing: predicate identification (PI) identifies words that evokes a semantic predicate; argument identification (AI) identifies the arguments of the predicates; argument classification (AC) labels the argument with the semantic relations (roles); and predicate classification (PC) further differentiate different use of the predicate word. All components are built as maximal entropy based classifiers, with their parameters estimated by the open source TADM system[4], feature sets selected on the development set. Evaluation results from previous years' CoNLL shared tasks show that the system achieves state-of-the-art performance, especially for its out-domain applications.

4 Textual Relatedness

As we mentioned in the introduction, we break down the three-way classification into a two-stage binary classification. Furthermore, we treat the first stage as a subtask of the main task, which determines whether H is relevant to T. Similar to the probabilistic entailment score, we use a relatedness score to measure such relationship. Due to the nature of the entailment recognition that H should be fully entailed by T, we also make this relatedness relationship asymmetric. Roughly speaking, this *Relatedness* function $R(T, H)$ can be described as whether or how relevant H is to some part of T. The relevance can be realized as string similarity, semantic similarity, or being co-occurred in similar contexts.

Before we define the relatedness function formally, let us look at the representation again. After semantic parsing described in the previous section, we obtain a PAS for each sentence. On top of it, we define a *predicate-argument graph* (PAG), the nodes of which are predicates, arguments or sometimes both, and the edges of which are labeled semantic relations. Notice that each predicate can dominate zero, one, or more arguments, and each argument have one or more predicates which dominate it. Furthermore, the graph is not necessarily fully connected. Thus, the $R(T, H)$ function can be defined on the dependency representation as follows: if the PAG of H is semantically relevant to part of the PAG of T, H is semantically relevant to T.

In order to compare the two graphs, we further reduce the alignment complexity by breaking the graphs into sets of trees. Two types of decomposed trees are considered: one is to take each predicate as the root of a tree and arguments as child nodes, and the other is on the contrary, to take each argument as root and their governing predicates as child nodes. We name them as *Predicate Trees* (P-Trees) and *Argument Trees* (A-Trees) respectively. To obtain the P-Trees, we enumerate each predicate, find all the arguments which it directly dominates, and then construct a P-Tree. The algorithm to obtain A-Trees works in the similar way. Finally, we will have a set of P-Trees and a set of A-Trees for each PAG, both of which are simple trees with depth of one. Figure 1 shows an example of such procedures. Notice that we do not consider cross-sentential inference, instead, we simply take the union of tree sets from all the sentences. Figure 2 illustrates the PAG for both T and H after semantic parsing, and the resulting P-Trees and A-Trees after applying the decomposition algorithm.

Formally, we define the relatedness function for a T-H pair as the maximum value of the relatedness scores of all pairs of trees in T and H (P-trees and A-trees).

[4] http://tadm.sourceforge.net/

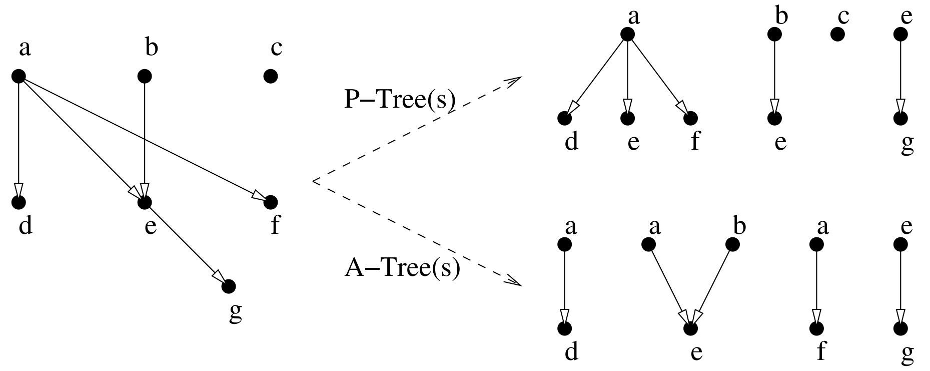

Figure 1: Decomposition of predicate-argument graphs (left) into P-Trees (right top) and A-Trees (right bottom)

$$R(T,H) = \max_{1\le i\le r, 1\le j\le s} \{R(Tree_{T_i}, Tree_{H_j})\}$$

In order to compare two P-Trees or A-Trees, we further define each predicate-argument pair contained in a tree as a semantic dependency triple. Each semantic dependency triple contains a predicate, an argument, and the semantic dependency label in between, in the form of $\langle Predicate, Dependency, Argument\rangle$. Then we define the relatedness function between two trees as the minimum value of the relatedness scores of all the triple pairs from the two trees.

$$R(Tree_T, Tree_H) = \min_{1\le i\le n, 1\le j\le m} \\ \{R(\langle P_T, D_{T_i}, A_{T_i}\rangle, \langle P_H, D_{H_j}, A_{H_j}\rangle)\}$$

For the relatedness function between two semantic dependency triples, we define the following two settings: the FULL match and the NOTFULL match. Either match requires that the predicates are related at the first place. The former means both the dependencies and the arguments are related; while the latter only requires the dependencies to be related.

$$R(\langle P_T, D_T, A_T\rangle, \langle P_H, D_H, A_H\rangle) = \\ \begin{cases} \text{Full} & R(P_T,P_H)=R(D_T,D_H)=R(A_T,A_H)=1 \\ \text{NotFull} & R(P_T,P_H)=R(D_T,D_H)=1 \\ \text{Other} & \text{Otherwise} \end{cases}$$

Now, the only missing components in our definition is the relatedness functions between predicates, arguments, and semantic dependencies. Fortunately, many people have done research on semantic relatedness in lexical semantics that we could use. Therefore, these functions can be realized by different string matching algorithms and/or lexical resources. Since the meaning of relevance is rather wide, apart from the string matching of the lemmas, we also incorporate various resources, from distributionally collected ones to hand-crafted ontologies. We choose VerbOcean (Chklovski and Pantel, 2004) to obtain the relatedness between predicates (after using WordNet (Fellbaum, 1998) to change all the nominal predicates into verbs) and use WordNet for the argument alignment. For the verb relations in VerbOcean, we consider all of them as related; and for WordNet, we not only use the synonyms, hyponyms, and hypernyms, but antonyms as well. Consequently, we simplify these basic relatedness functions into a binary decision. If the corresponding strings are matched or the relations mentioned above exist, the two predicates, arguments, or dependencies are related; otherwise, not.

In addition, the Normalized Google Distance (NGD) (Cilibrasi and Vitanyi, 2007) is applied to both cases[5]. As for the comparison between dependencies, we simply apply the string matching, except for modifier labels, which we treat them as the same[6]. In all, the main idea here is to incorporate both distributional semantics and ontological semantics in order to see whether their contributions are overlapping or complementary. In practice, we use empirical value 0.5 as the threshold. Below the threshold means they are related, oth-

[5] You may find the NGD values of all the content word pairs in RTE-3 and RTE-4 datasets at `http://www.coli.uni-sb.de/~rwang/resources/RTE3_RTE4_NGD.txt`.

[6] This is mainly because it is more difficult for the SRL system to differentiate modifier labels than the complements.

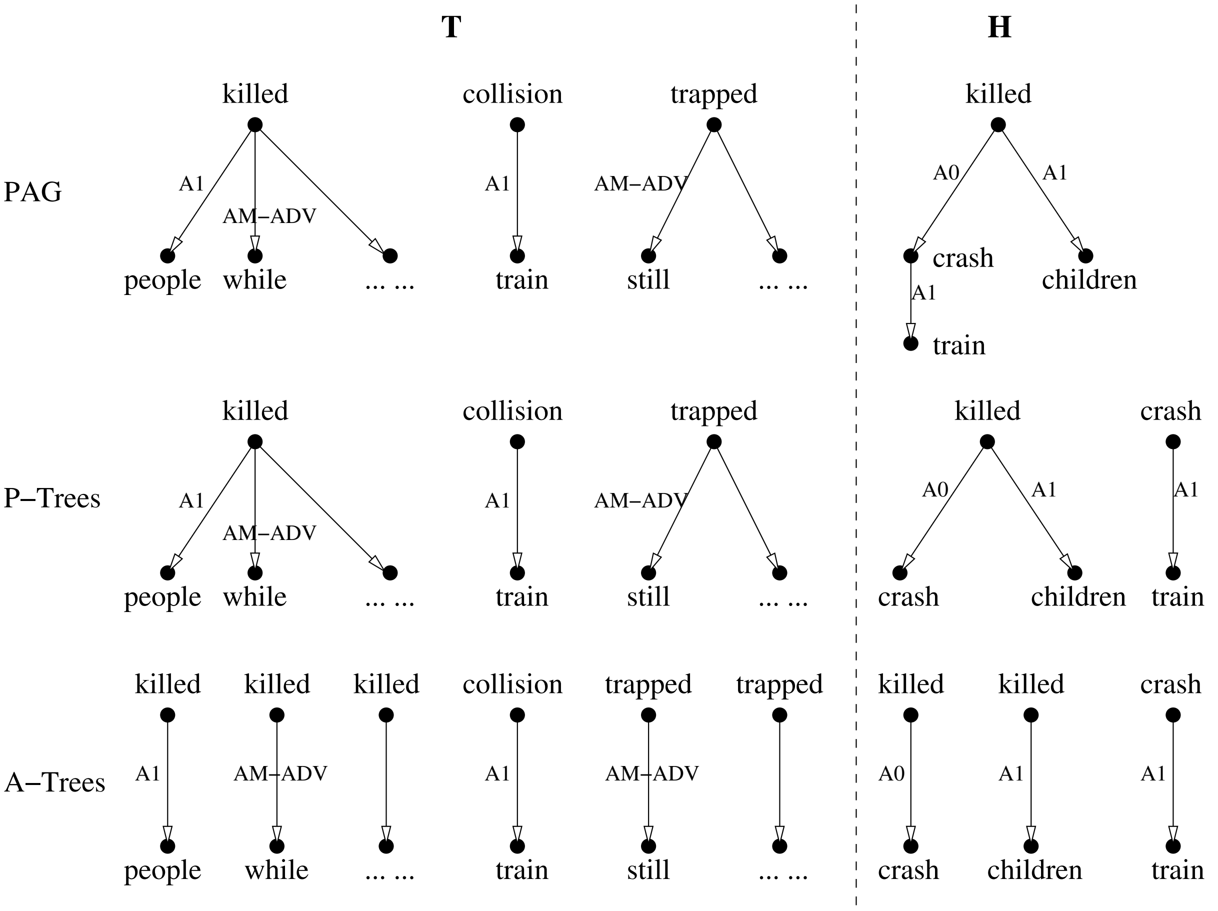

Figure 2: Predicate-argument graphs and corresponding P-Trees and A-trees of the T-H pair

erwise not. In order to achieve a better coverage, we use the OR operator to connect all the relatedness functions above, which means, if any of them holds, the two items are related.

Notice that, although we define only the relatedness between T and H, in principle, the graph representation can also be used for the entailment relationship. However, since it needs more fine-grained analysis and resources, we will leave it as the future work.

5 Experiments

In order to evaluate our method, we setup several experiments. The baseline system here is a simple Naive Bayes classifier with a feature set containing the Bag-of-Words (BoW) overlapping ratio between T and H, and also the syntactic dependency overlapping ratio. The feature model combines two baseline systems proposed by previous work, which gives out quite competitive performance. Since the main goal of this paper is to show the impact of the PAS-based alignment module, we will not compare our results with other RTE systems (In fact, the baseline system already outperforms the average accuracy score of the RTE-4 challenge).

The main data set used for testing here is the RTE-4 data set with three-way annotations (500 entailment T-H pairs (E), 150 contradiction pairs (C), and 350 unknown pairs (U)). The results on RTE-3 data set (combination of the development set and test set, in all, 822 E pairs, 161 C pairs, and 617 U pairs) is also shown, although the original annotation is two-way and the three-way annotation was done by different researchers after the challenge[7].

We will first show the performance of the baseline systems, followed by the results of our PAS-based alignment module and its impact on the whole task. After that, we will also give more detailed analysis of our alignment module, according to different lexical relatedness measurements.

[7] The annotation of the development set was done by students at Stanford, and the annotation of the test set was done as double annotation by NIST assessors, followed by adjudication of disagreements. Answers were kept consistent with the two-way decisions in the main task gold answer file.

5.1 Baselines

The baseline systems used here are based on overlapping ratio of words and syntactic dependencies between T and H. For the word overlapping ratio, we calculate the number of overlapping tokens between T and H and normalize it by dividing it by the number of tokens in H. The syntactic dependency overlapping ratio works similarly: we calculate the number of overlapping syntactic dependencies and divide it by the number of syntactic dependencies in H, i.e. the same as the number of tokens. Enlightened by the relatedness function, we also allow either FULL match (meaning both the dependencies and the parent tokens are matched), and NOTFULL match (meaning only the dependencies are matched). Here we only use string match between lemmas and syntactic dependencies. Table 1 presents the performance of the baseline system.

The results show that, even with the same classifier and the same feature model, with a proper two-stage strategy, it can already achieve better results than the three-way classification. Note that, the first strategy is not so successful, and that is the traditional two-way annotation of the RTE task. Our explanation here is that the BoW method (even with syntactic dependency features) is based on overlapping information shared by T and H, which essentially means the more information they share, the more relevant they are, instead of being more similar or the same. Therefore, for the "$ECU \rightarrow E/CU$" setting, methods based on overlapping information are not the best choice, while for "$ECU \rightarrow U/EC$", they are more appropriate.

In addition, the upper bound numbers show the accuracy when the first-stage classification is perfect, which give us an indication of how far we could go. The lower upper bound for the second strategy is mainly due to the low proportion of the C cases (15%) in the data set; while the other two both show large space for improvement.

5.2 The PAS-based Alignment Module

In this subsection, we present a separate evaluation of our PAS-based alignment module. As we mentioned before (cf. Section 4), there are several parameters to be tuned in our alignment algorithm: a) whether the relatedness function between P-Trees asks for the FULL match; b) whether the function for A-Trees asks for the FULL match; and c) whether both P-Trees and A-Trees being related are required or either of them holds is enough. Since they are all binary values, we use the 3-digit code to represent each setting, e.g. [FFO][8] means *either* P-Trees are FULL matched *or* A-Trees are FULL matched. The performances of different settings of the module are shown in the following Precision-Recall figure 3,

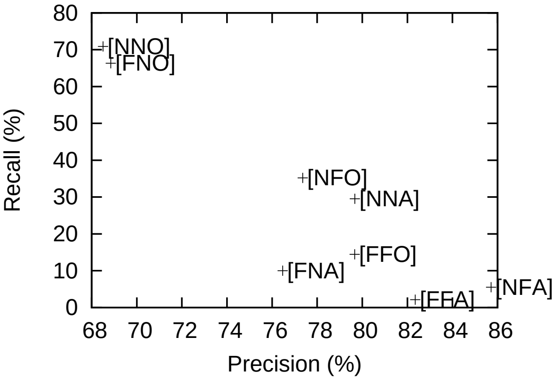

Figure 3: Precision and recall of different alignment settings

Since we will combine this module with the baseline system and it will be integrated as the first-stage classification, the F1 scores are not indicative for selecting the best setting. Intuitively, we may prefer higher precision than recall.

One limitation of our method we need to point out here is that, if some important predicates or arguments in H are not (correctly) identified by the SRL system, fewer P-Trees and A-Trees are required to be related to some part of T, thus, the relatedness of the whole pair could easily be satisfied, leading to false positive cases.

5.3 Impact on the Final Results

The best settings for RTE-3 data set is [NNA] and for RTE-4 data set is [NFO], which are both in the middle of the setting range shown in the previous figure 3.

As for the integration of the PAS-based alignment model with our BoW-based baseline, we only consider the third two-stage classification strategy in Table 1. Other strategies would also be interesting to try, however, the proposed alignment algorithm exploits relatedness between T and H, which might not be fine-grained enough to detect

[8] F stands for FULL, and O stands for OR. Other letters are, N stands for NOTFULL, and A stands for AND.

Strategies	Three-Way	Two-Stage		
	$E/C/U$	$E/CU \rightarrow E/C/U$	$C/EU \rightarrow C/E/U$	$U/EC \rightarrow U/E/C$
Accuray	53.20%	50.00%	53.50%	54.20%
Upper Bound	/	82.80%	68.70%	84.90%

Table 1: Performances of the Baselines

entailment or contradiction. New alignment algorithm has to be designed to explore other strategies. Thus, in this work, we believe that the alignment algorithm based on PAS (and other methods based on overlapping information between T and H) is suitable for the $U/EC \rightarrow U/E/C$ classification strategy.

Table 2 shows the final results.

The first observation is that the improvement of accuracy on the first stage of the classification can be preserved to the final results. And our PAS-based alignment module can help, though there is still large space for improvement. Compared with the significantly improved results on RTE-4, the improvement on RTE-3 is less obvious, mainly due to the relatively lower precision (70.33% vs. 79.67%) of the alignment module itself.

Also, we have to say that the improvement is not as big as we expected. There are several reasons for this. Besides the limitation of our approach mentioned in the previous section, the predicates and arguments themselves might be too sparse to convey all the information we need for the entailment detection. In addition, in some sense, the baseline is quite strong for this comparison, since the PAS-based alignment module relies on the overlapping words at the first place, there are quite a few pairs solved by both the main approach and the baseline. Then, it would be interesting to take a closer look at the lexical resources used in the main system, which is another additional knowledge it has, comparing with the baseline.

5.4 Impact of the Lexical Resources

We did an ablation test of the lexical resources used in our alignment module. Recall that we have applied three lexical resources, VerbOcean for the predicate relatedness function, WordNet for the argument relatedness function, and Normalized Google Distance for both. Table 3 shows the performances of the system without each of the resources,

The results clearly show that each lexical resource does contribute some improvement to the final performance of the system and it confirms the idea of combining lexical resources being acquired in different ways. For instance, at the beginning, we expected that the relationship between "*people*" and "*children*" could be captured by WordNet, but in fact not. Fortunately, the NGD has a quite low value of this pair of words (0.21), which suggests that they occur together quite often, or in other words, they are relevant.

One interesting future work on this aspect is to substitute the OR connector between these lexical resources with an AND operator. Thus, instead of using them to achieve a higher coverage, whether they could be filters for each other to increase the precision will also be interesting to know.

6 Conclusion and Future Work

In this paper, we address the motivation and issues of casting the three-way RTE problem into a two-stage binary classification task. We apply an SRL system to derive the predicate-argument structure of the input sentences, and propose ways of calculating semantic relatedness between the shallow semantic structures of T and H. The experiments show improvements in the first-stage classification, which accordingly contribute to the final results of the RTE task.

For future work, we would like to see whether the PAS can help the second-stage classification as well, e.g. the semantic dependency of negation (AM-NEG) could be helpful for the contraction recognition. Furthermore, since the PAS is usually a bag of unconnected graphs, we could find a way to joint them together, in order to consider both inter- and intra- sentential inferences based on it.

In addition, this approach has the potential to be integrated with other RTE modules. For instance, for the predicate alignment, we may consider to use DIRT rules (Lin and Pantel, 2001) or other paraphrase resources (Callison-Burch, 2008), and for the argument alignment, external named-entity recognizer and anaphora resolver would be very helpful. Even more, we also plan to compare/combine it with other methods which are not based on overlapping information between T and H.

Systems	Baseline1	Baseline2	SRL+Baseline2	The First Stage		
Data Sets	Three-Way	Two-Stage	Two-Stage	Baseline2	SRL+Baseline2	SRL
RTE-3 [NNA]	52.19%	52.50%	53.69%(2.87%↑)	59.50%	60.56%(1.78%↑)	70.33%
RTE-4 [NFO]	53.20%	54.20%	56.60%(6.39%↑)	67.10%	70.20%(4.62%↑)	79.67%

Table 2: Results on the Whole Datasets

Data Sets	SRL+Baseline	SRL+Baseline - VO	SRL+Baseline - NGD	SRL+Baseline - WN
RTE-3 [NNA]	53.69%	53.19%(0.93%↓)	53.50%(0.35%↓)	52.88%(1.51%↓)
RTE-4 [NFO]	56.60%	56.00%(1.06%↓)	56.10%(0.88%↓)	55.70%(1.59%↓)

Table 3: Impact of the Lexical Resources

Acknowledgments

The first author is funded by the PIRE PhD scholarship program sponsored by the German Research Foundation (DFG). The second author thanks the German Excellence Cluster of Multimodal Computing and Interaction for the support of the work.

References

Eugene Agichtein, Walt Askew, and Yandong Liu. 2009. Combining Lexical, Syntactic, and Semantic Evidence for Textual Entailment Classification. In *Proceedings of the First Text Analysis Conference (TAC 2008)*.

Collin F. Baker, Charles J. Fillmore, and John B. Lowe. 1998. The berkeley framenet project. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, Volume 1*, pages 86–90, Montreal, Canada.

Aljoscha Burchardt, Nils Reiter, Stefan Thater, and Anette Frank. 2007. A semantic approach to textual entailment: System evaluation and task analysis. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, Prague, Czech Republic.

Chris Callison-Burch. 2008. Syntactic constraints on paraphrases extracted from parallel corpora. In *Proceedings of EMNLP*.

Timothy Chklovski and Patrick Pantel. 2004. VerbOcean: Mining the Web for Fine-Grained Semantic Verb Relations. In *Proceedings of Conference on Empirical Methods in Natural Language Processing (EMNLP-04)*, Barcelona, Spain.

Rudi Cilibrasi and Paul M. B. Vitanyi. 2007. The Google Similarity Distance. *IEEE/ACM Transactions on Knowledge and Data Engineering*, 19(3):370–383.

Peter Clark and Phil Harrison. 2009. Recognizing Textual Entailment with Logical Inference. In *Proceedings of the First Text Analysis Conference (TAC 2008)*.

Ido Dagan, Oren Glickman, and Bernardo Magnini. 2006. The PASCAL Recognising Textual Entailment Challenge. In *Machine Learning Challenges*, volume 3944 of *Lecture Notes in Computer Science*, pages 177–190. Springer.

Marie-Catherine de Marneffe, Anna N. Rafferty, and Christopher D. Manning. 2008. Finding contradictions in text. In *Proceedings of ACL-08*.

Christiane Fellbaum. 1998. *WordNet: An Electronic Lexical Database*. MIT Press.

Danilo Giampiccolo, Hoa Trang Dang, Bernardog Magnini, Ido Dagan, Elena Cabrio, and Bill Dolan. 2009. The Fourth PASCAL Recognizing Textual Entailment Challenge. In *Proceedings of the First Text Analysis Conference (TAC 2008)*.

Jan Hajič, Massimiliano Ciaramita, Richard Johansson, Daisuke Kawahara, Maria Antònia Martí, Lluís Màrquez, Adam Meyers, Joakim Nivre, Sebastian Padó, Jan Štěpánek, Pavel Straňák, Mihai Surdeanu, Nianwen Xue, and Yi Zhang. 2009. The CoNLL-2009 shared task: Syntactic and semantic dependencies in multiple languages. In *Proceedings of the 13th Conference on Computational Natural Language Learning*, Boulder, CO, USA.

Sanda Harabagiu and Andrew Hickl. 2006. Methods for Using Textual Entailment in Open-Domain Question Answering. In *Proceedings of COLING-ACL 2006*, pages 905–912, Sydney, Australia.

Dekang Lin and Patrick Pantel. 2001. DIRT - Discovery of Inference Rules from Text. In *In Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining*.

Bill MacCartney, Michel Galley, and Christopher D. Manning. 2008. A phrase-based alignment model for natural language inference. In *Proceedings of EMNLP 2008*.

Ryan McDonald, Fernando Pereira, Kiril Ribarov, and Jan Hajic. 2005. Non-Projective Dependency Parsing using Spanning Tree Algorithms. In *Proceedings of hlt-emnlp 2005*, pages 523–530, Vancouver, Canada.

Sebastian Pado, Marie-Catherine de Marneffe, Bill MacCartney, Anna N. Rafferty, Eric Yeh, and

Christopher D. Manning. 2009. Deciding entailment and contradiction with stochastic and edit distance-based alignment. In *Proceedings of the First Text Analysis Conference (TAC 2008)*.

Reda Siblini and Leila Kosseim. 2009. Using Ontology Alignment for the TAC RTE Challenge. In *Proceedings of the First Text Analysis Conference (TAC 2008)*.

Mihai Surdeanu, Richard Johansson, Adam Meyers, Lluís Màrquez, and Joakim Nivre. 2008. The CoNLL-2008 shared task on joint parsing of syntactic and semantic dependencies. In *Proceedings of the 12th conference on computational natural language learning (CoNLL-2008)*, Manchester, UK.

Rui Wang and Günter Neumann. 2007. Recognizing textual entailment using a subsequence kernel method. In *Proceedings of the Twenty-Second AAAI Conference on Artificial Intelligence (AAAI-07)*, pages 937–942, Vancouver, Canada.

Yi Zhang, Rui Wang, and Hans Uszkoreit. 2008. Hybrid Learning of Dependency Structures from Heterogeneous Linguistic Resources. In *Proceedings of the Twelfth Conference on Computational Natural Language Learning (CoNLL 2008)*, pages 198–202, Manchester, UK.