Weakly Supervised Learning of Presupposition Relations between Verbs

Galina Tremper
Department of Computational Linguistics
Heidelberg University, Germany
tremper@cl.uni-heidelberg.de

Abstract

Presupposition relations between verbs are not very well covered in existing lexical semantic resources. We propose a weakly supervised algorithm for learning presupposition relations between verbs that distinguishes five semantic relations: presupposition, entailment, temporal inclusion, antonymy and other/no relation. We start with a number of seed verb pairs selected manually for each semantic relation and classify unseen verb pairs. Our algorithm achieves an overall accuracy of 36% for type-based classification.

1 Introduction

A main characteristics of natural language is that significant portions of content conveyed in a message may not be overtly realized. This is the case for presuppositions: e.g, the utterance *Columbus didn't manage to reach India.* presupposes that *Columbus had tried to reach India.* This presupposition does not need to be stated, but is implicitly understood. Determining the presuppositions of events reported in texts can be exploited to improve the quality of many natural language processing applications, such as information extraction, text understanding, text summarization, question-answering or machine translation.

The phenomenon of presupposition has been throughly investigated by philosophers and linguists (i.a. Stalnaker, 1974; van der Sandt, 1992). There are only few attempts for practical implementations of presupposition in computational linguistics (e.g. Bos, 2003). Especially, presupposition is understudied in the field of corpus-based learning of semantic relations. Machine learning methods have been previously applied to determine semantic relations such as *is-a* and *part-of*, also *succession*, *reaction* and *production* (Pantel and Pennacchiotti, 2006). Chklovski and Pantel (2004) explored classification of fine-grained verb semantic relations, such as *similarity*, *strength*, *antonymy*, *enablement* and *happens-before*. For the task of entailment recognition, learning of entailment relations was attempted (Pekar, 2008). None of the previous work investigated subclassifying semantic relations including presupposition and entailment, two relations that are closely related, but behave differently in context.

In particular, the inferential behaviour of presuppositions and entailments crucially differs in special semantic contexts. E.g., while presuppositions are preserved under negation (as in *Columbus managed/didn't manage to reach India* the presupposition *tried to*), entailments do not survive under negation (*John F. Kennedy has been/has not been killed*). Here the entailment *died* only survives in the positive sentence. Such differences are crucial for both analysis and generation-oriented NLP tasks.

This paper presents a weakly supervised algorithm for learning presupposition relations between verbs cast as a discriminative classification problem. The structure of the paper is as follows: Section 2 reviews state of the art. Section 3 introduces our task and the learning algorithm. Section 4 reports on experiment organization; the results are presented in Section 5. Finally, we summarise and present objectives for future work.

2 Related Work

One of the existing semantic resources related to our paper is WordNet (Fellbaum, 1998). It comprises lexical semantic information about English nouns, verbs, adjectives and adverbs. Among the semantic relations defined specifically for verbs are entailment, hyponymy, troponymy, antonymy and cause. However, not all of them are well covered, for example, there are only few entries for presupposition and entailment in WordNet.

Proceedings of the ACL 2010 Student Research Workshop, pages 97–102,
Uppsala, Sweden, 13 July 2010. ©2010 Association for Computational Linguistics

One attempt to acquire fine-grained semantic relations from corpora is VerbOcean (Chklovski and Pantel, 2004). Chklovski and Pantel used a semi-automatic approach for extracting semantic relations between verbs using a list of patterns. The selection of the semantic relations was inspired by WordNet. VerbOcean showed good accuracy values for the *antonymy* (50%), *similarity* (63%) and *strength* (75%) relations. However, VerbOcean doesn't distinguish between *entailment* and *presupposition*; they are conflated in the classes *enablement* and *happens-before*.

A distributional method for extracting highly associated verbs was proposed by Lin and Pantel (2001). This method extracts semantically related words with good precision, but it does not determine the type and symmetry of the relation. However, the method is able to recognize the existence of semantic relations holding between verbs and hence can be used as a basis for finding and further discriminating more detailed semantic relations.

3 A Weakly Supervised Approach to Learning Presupposition Relations

We describe a weakly supervised approach for learning semantic relations between verbs including implicit relations such as presupposition. Our aim is to perform a type-based classification of verb pairs. I.e., we determine the class of a verb-pair relation by observing co-occurrences of these verbs in contexts that are indicative for their intrinsic meaning relation. This task differs from a token-based classification, which aims at classifying each verb pair instance as it occurs in context.

Classified relations. We distinguish between the five classes of semantic relations presented in Table 1. We chose *entailment*, *temporal inclusion* and *antonymy*, because these relations may be confounded with the *presupposition* relation. A special class *other/no* comprises semantic relations not discussed in this paper (e.g. synonymy) and verb pairs that are not related by a semantic relation. The relations can be subdivided into symmetric and asymmetric relations, and relations that involve temporal sequence, or those that do not involve a temporal order, as displayed in Table 1.

Semantic Relation	Example	Symmetry	Temporal Sequence
Presupposition	*find - seek,* *answer - ask*	asymmetric	yes
Entailment	*look - see,* *buy - own*	asymmetric	yes
Temporal Inclusion	*walk - step,* *talk - whisper*	symmetric	no
Antonymy	*win - lose,* *love - hate*	symmetric	no
Other/no	*have - own,* *sing - jump*	undefined	undefined

Table 1: Selected Semantic Relations

A Weakly Supervised Learning Approach. Our algorithm starts with a small number of seed verb pairs selected manually for each relation and iteratively classifies a large set of unseen and unlabeled verb pairs. Each iteration has two phases:

1. **Training the Classifiers** We independently train binary classifiers for each semantic relation using both shallow and deep features.
2. **Ensemble Learning and Ranking** Each of the five classifiers is applied to each sentence from an unlabeled corpus. The predictions of the classifiers are combined using ensemble learning techniques to determine the most confident classification. The obtained list of the classified instances is ranked using pattern scores, in order to select the most reliable candidates for extension of the training set.

Features. Both shallow lexical-syntactic and deep syntactic features are used for the classification of semantic relations. They include:

1. the distance between two analyzed verbs and the order of their appearance
2. verb form (tense, aspect, modality, voice), presence of negation and polarity verbs[1]
3. coordinating/subordinating conjunctions
4. adverbial adjuncts
5. PoS-tag-contexts (two words preceding and two words following each verb)
6. the length of the path of grammatical functions relating the two verbs
7. co-reference relation holding between the subjects and objects of the verbs (both verbs have the same subject/object, subject of one verb corresponds to the object of the second or there is no relation between them).

In order to extract these features the training corpus is parsed using a deep parser.

[1] Polarity verbs are taken from the polarity lexicon of Nairn et al. (2006). It encodes whether the complement of proposition embedding verbs is true or false. We used the verbs themselves as a feature without their polarity-tags.

4 Experimental Setting

Initial Subset of Verb Pair Candidates. Unlike other semi-supervised approaches, we don't use patterns for acquiring new candidates for classification. Candidate verb pairs are obtained from a previously compiled list of highly associated verbs. We use the DIRT Collection (Lin and Pantel, 2001) from which we further extract pairs of highly associated verbs as candidates for classification. The advantage of this resource is that it consists of pairs of verbs which stand in a semantic relation (cf. Section 2). This considerably reduces the number of verb pairs that need to be processed as candidates in our classification task.

DIRT contains 5,604 verb types and 808,764 verb pair types. This still represents a huge number of verb pairs to be processed. We therefore filtered the extracted set by checking verb pair frequency in the first three parts of the ukWAC corpus (Baroni et al., 2009) (UKWAC_1. . . 3) and by applying the PMI test with threshold 2.0. This reduces the number of verb pairs to 199,393.

For each semantic relation we select three verb pairs as seeds. The only exception is *temporal inclusion* for which we selected six verb pairs, due to the low frequency of such verb pairs within a single sentence. These verb pairs were used for building an initial training corpus of verb pairs in context. The remaining verb pairs are used to build the corpus of unlabeled verb pairs in context in the iterative classification process.

Preprocessing. Given these verb pairs, we extracted sentences for training and for unlabeled data set from the first three parts of the UKWAC corpus (Baroni et al., 2009). We compiled a set of CQP queries (Evert, 2005) to find sentences that contain both verbs of a verb pair and applied them on UKWAC_1. . . 3 to build the training and unlabeled subcorpora. We filter out sentences with more than 60 words and sentences with a distance between verbs exceeding 20 words. To avoid growing complexity, only sentences with exactly one occurrence of each verb pair are retained. We also remove sentences that trigger wrong candidates, in which the auxiliaries *have* or *do* appear in a candidate verb pair.

The corpus is parsed using the XLE parser (Crouch et al., 2008). Its output contains both the structural and functional information we need to extract the shallow and deep features used in the classification, and to generate patterns.

Training Corpus. From this preprocessed corpus, we created a training corpus that contains three different components:

1. *Manually annotated training set.* All sentences containing seed verb pairs extracted from UKWAC_1 are annotated manually with two values *true/false* in order to separate the negative training data.
2. *Automatically annotated training set.* We build an extended, heuristically annotated training set for the seed verb pairs, by extracting further instances from the remaining corpora (UKWAC_2 and UKWAC_3). Using the manual annotations of step 1., we manually compiled a small stoplist of patterns that are used to filter out wrong instances. The constructed stoplist serves as an elementary disambiguation step. For example, the verbs *look* and *see* can stand in an entailment relation if *look* is followed by the prepositions *at, on, in*, but not in case of prepositions *after* or *forward* (e.g. *looking forward to*).
3. *Synonymous verb pairs.* To further enrich the training set of data, synonyms of the verb pairs are manually selected from WordNet. The corresponding verb pairs were extracted from UKWAC_1. . . 3. In order to avoid adding noise, we used only synonyms of unambiguous verbs. The problem of ambiguity of the target verbs wasn't considered at this step.

The overall size of the training set for the first classification step is 15,717 sentences from which 5,032 are manually labeled, 9,918 sentences are automatically labeled and 757 sentences contain synonymous verb pairs. The distribution is unbalanced: *temporal inclusion* e.g. covers only 2%, while *entailment* covers 39% of sentences. We balanced the training set by undersampling *entailment* and *other/no* by 20% and correspondingly oversampling the *temporal inclusion* class.

Patterns. Similar to other pattern-based approaches we use a set of seed verb pairs to induce indicative patterns for each semantic relation. We use the induced patterns to restrict the number of the verb pair candidates and to rank the labelled instances in the iterative classification step.

The patterns use information about the verb forms of analyzed verb pairs, modal verbs and the polarity verbs (only if they are related to the analyzed verbs) and coordinating/subordinating conjunctions connecting two verbs. The analyzed verbs in the sentence are substituted with V1 and V2 placeholders in the pattern. For example, for the sentence: *Here we should be careful for there are those who seek and do not find.* and the verb pair *(find,seek)* we induce the following pattern: *V2 and do* [*not*|*n't*] *V1*. The patterns are extracted automatically from deep parses of the training corpus. Examples of the best patterns we determined for semantic relations are presented in Table 2.

Semantic Relation	Patterns
Presupposition	V2-ed * though * was * V1-ed, V2-ed * but was [not|n't] V1-ed, V2-ing * might V1
Entailment	if * V1 * V2, V1-ing * [shall|will|'ll] V2, V2 * by V1-ing
Temporal Inclusion	V2 * V1-ing, V1-ing and V2-ing, when V2 * V1
Antonymy	V1 or * V2, either * V1 or * V2, V1-ed * but V2-ed
Other/no	V1 * V2, V1-ing * V2-ing, V2-ed * and * V1-ed

Table 2: Patterns for Selected Semantic Relations

Pattern ranks are used to compute the reliability score for instances, as proposed by Pantel and Pennacchiotti (2006). The pattern reliability is calculated as follows:

$$r_\pi(p) = \frac{1}{|I|} \sum_{i \in I} \frac{pmi(i,p)}{max_{pmi}} \times r_i(i) \quad (1)$$

where:
$pmi(i,p)$ - pointwise mutual information (PMI) between the instance i and the pattern p;
max_{pmi} - maximum PMI between all patterns and all instances;
$r_i(i)$ - reliability of an instance i. For seeds $r_i(i) = 1$ (they are selected manually), for the next iterations the instance reliability is:

$$r_i(i) = \frac{1}{|P|} \sum_{p \in P} \frac{pmi(i,p)}{max_{pmi}} \times r_\pi(p) \quad (2)$$

We also consider using the patterns as a feature for classification, in case they turn out to be sufficiently discriminative.

Training Binary Classifiers. We independently train 5 binary classifiers, one for each semantic relation, using the J48 decision tree algorithm (Witten and Frank, 2005).

Data Sets. As the primary goal of this paper is to classify semantic relations on the type level, we elaborated a first gold standard dataset for type-based classification. We used a small sample of 100 verb pairs randomly selected from the automatically labeled corpus. This sample was manually annotated by two judges after we had eliminated the system annotations in order not to influence the judges' decisions. The judges had the possibility to select more than one annotation, if necessary. We measured inter-annotator agreement was 61% ($k \approx 0.21$). The low agreement shows the difficulty of decision in the annotation of fine-grained semantic relations.[2]

While the first gold standard dataset of verb pairs was annotated *out of context*, we constructed a second gold standard of verb pairs annotated at the token level, i.e. in context. This second data set can be used to evaluate a token-based classifier (a task not attempted in the present paper). It also offers a ground truth for type-based classification, in that it controls for contextual ambiguity effects. I.e., we can extract a type-based gold standard on the basis of the token-annotated data.[3] We proposed to one judge to annotate the same 100 verb pair types as in the previous annotation task, this time in context. For this purpose we randomly selected 10 instances for each verb pair type (for rare verb pair types only 5). We compared the gold standards elaborated by the same judge for type-based and token-based classification:

- 62% of verb pair types were annotated with the same labels on both levels, indicating correct annotation
- 10% of verb pair types were assigned conflicting labels, indicating wrong annotation
- 28% of verb pair types were assigned labels not present on the type level, or the type level label was not assigned in context

The figures show that for the most part the type-based annotation conforms with the ground truth obtained from token-based annotation. Only 10% of verb pair types were established as conflicting with the ground truth. The remaining 28% can be considered as potentially correct: either the annotated data does not contain the appropriate context for a given type label or the type-level anno-

[2] Data inspection revealed that one annotator was more experienced in semantic annotation tasks. We evaluate our system using the annotations of only one judge.

[3] This option was not pursued in the present paper.

tation, performed without context, does not foresee an existing relation. This points to a general difficulty, namely to acquire representative data sets for token-level annotation, and also to perform type-level annotations without context for the present task.

Combining Classifiers in Ensemble Learning. Both token-based and type-based classification starts with determining of the most confident classification for instances. Each instance of the corpus of unlabeled verb pairs is classified by the individual binary classifiers. In order to select the most confident classification we compare the votes of the individual classifiers as follows:

1. If an instance is classified by one of the classifiers as *true* with confidence less than 0.75, we discard this classification.
2. If an instance is classified as *true* by more than one classifier, we consider only the classification with the highest confidence.[4]

In contrast to token-based classification that accepts only one semantic relation, for type-based classification we allow the existence of more than one semantic relation for a verb pair. To avoid the unreliable classifications, we apply several filters:

1. If less than 10% of the instances for a verb pair are classified with some specific semantic relation, this classification is considered to be unconfident and is discarded.
2. If a verb pair is classified as positive for more than three semantic relations, this verb pair remains unclassified.
3. If a verb pair is classified with up to three semantic relations and if more than 10% of the examples are classified with any of these relations, the verb pair is labeled with all of them.

Iteration and Stopping Criterion. After determining the most confident classification we rank the instances, following the ranking procedure of Pantel and Pennacchiotti (2006). Instances that exceed a reliability threshold (0.3 for our experiment) are selected for the extended training set. The remainining instances are returned to the unlabeled set. The algorithm stops if the average reliability score is smaller than a threshold value. In our paper we concentrate on the first iteration. Extension of the training set and re-ranking of patterns will be reported in future work.

Semantic relation (Count1/Count2)	Majority	Without NONE	Baseline
Presupposition (12/22)	67%	36%	18%
Entailment (9/20)	67%	35%	8%
Temp. Inclusion (7/11)	71%	36%	19%
Antonymy (11/24)	72%	42%	12%
NONE (61/29)	49%	31%	43%
Macro-Average	56%	36%	
Micro-Average	65%	36%	

Table 3: Accuracy for type-based classification

5 Evaluation Results

Results for type-based classification. We evaluate the accuracy of classification based on two alternative measures:

1. *Majority* - the semantic relation with which the majority of the sentences containing a verb pair have been annotated.
2. *Without NONE* - as in 1., but after removing the label *NONE* from all relation assignments except for those cases where NONE is the only label assigned to a verb pair.[5]

We computed accuracy as the number of verb pairs which were correctly labeled by the system divided by the total number of system labels. We compare our results against a baseline of random assignment, taking the distribution found in the manually labeled gold standard as the underlying verb relation distribution. Table 3 shows the accuracy results for each semantic relation[6].

Results for token-based classification. We also evaluate the accuracy of classification for token-based classification as the number of instances which were correctly labeled by the system divided by the total number of system labels. As the baseline we took the relation distribution on the token level. Table 4 shows the accuracy results for each semantic relation.

Discussion. The results obtained for type-based classification are well above the baseline with one exception. The best performance is achieved by *antonymy* (72% and 42% respectively for both

[4] We assume that within a given context a verb pair can exhibit only one relation.

[5] The second measure was used because in many cases the relation NONE has been determined to be the majority class.

[6] Count1 is the total number of system labels for the Majority measure and Count2 is the total number of system labels for the Without NONE measure.

Semantic relation	Count	Accuracy	Baseline
Presupposition	43	21%	8%
Entailment	39	15%	5%
Temp. Inclusion	15	13%	3%
Antonymy	34	29%	5%
NONE	511	81%	79%
Macro-Average		61%	
Micro-Average		31%	

Table 4: Accuracy for token-based classification

measures), followed by *temporal inclusion*, *presupposition* and *entailment*. Accuracy scores for token-based classification (excluding NONE) are lower at 29% to 13%. Error analysis of randomly selected false positives shows that the main reason for lower accuracy on the token level is that the context is not always significant enough to determine the correct relation.

Comparison to Related Work. Other projects such as VerbOcean (Chklovski and Pantel, 2004) report higher accuracy: the average accuracy is 65.5% if at least one tag is correct and 53% for the correct preferred tag. However, we cannot objectively compare the results of VerbOcean to our system because of the difference in the set of relation classes and evaluation procedures. Similar to us, Chklovski and Pantel (2004) evaluated VerbOcean using a small sample of data which was presented to two judges for manual evaluation. In contrast to our setup, they didn't remove the system annotations from the evaluation data set. Given the difficulty of the classification we suspect that correction of system output relations for establishing a gold standard bears a strong risk in favouring system classifications.

6 Conclusion and Future Work

The results achieved in our experiment show that weakly supervised methods can be applied for learning presupposition relations between verbs. Our work also shows that they are more difficult to classify than other typical lexical semantic relations, such as antonymy. Error analysis suggests that many errors can be avoided if verbs are disambiguated in context. It would be interesting to test our algorithm with different amounts of manually annotated training sets and different combinations of manually and automatically annotated training sets to determine the minimal amount of data needed to assure good accuracy.

In future work we will integrate word sense disambiguation as well as information about predicate-argument structure. Also, we are going to analyze the influence of single features on the classification and determining optimal feature sets, as well as the question of including patterns in the feature set. In this paper we used the same combination of features for all classifiers.

7 Acknowledgements

I would like to thank Anette Frank for supervision of this work, Dekang Lin and Patrick Pantel for sharing the DIRT resource and Carina Silberer and Christine Neupert creation of the gold standard.

References

Baroni, M., Bernardini, S., Ferraresi, A., Zanchetta, E.: The WaCky Wide Web: a collection of very large linguistically processed web-crawled corpora. Journal of Language Resources and Evaluation, Vol.43 (3), 209–226 (2009)

Bos, J.: Implementing the Binding and Accommodation Theory for Anaphora Resolution and Presupposition Projection. Computational Linguistics, Vol.29 (2), 179–210 (2003)

Chklovski, T., Pantel, P.: Verbocean: Mining the web for fine-grained semantic verb relations. Proceedings of EMNLP 2004, 33–40, Barcelona (2004)

Crouch, D., Dalrymple, M., Kaplan, R., King, T., Maxwell, J., Newman, P.: XLE Documentation. Palo Alto Research Center (2008)

Evert, S.: The CQP Query Language Tutorial (CWB Version 2.2.b90). IMS, Stuttgart (2005)

Fellbaum, C.: WordNet: An Electronic Lexical Database. 1st edition, MIT Press (1998)

Lin, D., Pantel, P.: Discovery of Inference Rules for Question Answering. Natural Language Engineering, Vol.7, 343–360 (2001)

Nairn, R., Condoravdi, C., Karttunen, L.: Computing Relative Polarity for Textual Inference. Proc. of ICoS-5, Buxton, UK (2006)

Pantel, P., Pennacchiotti, M.: Espresso: Leveraging Generic Patterns for Automatically Harvesting Semantic Relations. COLING 2006, 113-120 (2006)

Pekar, V.: Discovery of event entailment knowledge from text corpora. Computer Speech & Language, Vol.22 (1), 1–16 (2008)

Stalnaker, R.C.: Pragmatic Presuppositions. Semantics and Philosophy, New York: Univ. Press (1974)

van der Sandt, R.: Presupposition Projection as Anaphora Resolution. Journal of Semantics, Vol.9, 333–377 (1992)

Witten, I., Frank, E.: Data Mining: Practical Machine Learning Tools and Techniques. (2005)