FBK-IRST: Kernel Methods for Semantic Relation Extraction

Claudio Giuliano and **Alberto Lavelli** and **Daniele Pighin** and **Lorenza Romano**
FBK-IRST, Istituto per la Ricerca Scientifica e Tecnologica
I-38050, Povo (TN), ITALY
{giuliano,lavelli,pighin,romano}@itc.it

Abstract

We present an approach for semantic relation extraction between nominals that combines shallow and deep syntactic processing and semantic information using kernel methods. Two information sources are considered: (i) the whole sentence where the relation appears, and (ii) WordNet synsets and hypernymy relations of the candidate nominals. Each source of information is represented by kernel functions. In particular, five basic kernel functions are linearly combined and weighted under different conditions. The experiments were carried out using support vector machines as classifier. The system achieves an overall F_1 of 71.8% on the Classification of Semantic Relations between Nominals task at SemEval-2007.

1 Introduction

The starting point of our research is an approach for identifying relations between named entities exploiting only shallow linguistic information, such as tokenization, sentence splitting, part-of-speech tagging and lemmatization (Giuliano et al., 2006). A combination of kernel functions is used to represent two distinct information sources: (i) the global context where entities appear and (ii) their local contexts. The whole sentence where the entities appear (*global context*) is used to discover the presence of a relation between two entities. Windows of limited size around the entities (*local contexts*) provide useful clues to identify the roles played by the entities within a relation (e.g., agent and target of a gene interaction). In the task of detecting *protein-protein* interactions, we obtained state-of-the-art results on two biomedical data sets. In addition, promising results have been recently obtained for relations such as *work for* and *org based in* in the news domain[1].

In this paper, we investigate the use of the above approach to discover semantic relations between nominals. In addition to the original feature representation, we have integrated deep syntactic processing of the global context and semantic information for each candidate nominals using WordNet as external knowledge source. Each source of information is represented by kernel functions. A tree kernel (Moschitti, 2004) is used to exploit the deep syntactic processing obtained using the Charniak parser (Charniak, 2000). On the other hand, bag of synonyms and hypernyms is used to enhance the representation of the candidate nominals. The final system is based on five basic kernel functions (bag-of-words kernel, global context kernel, tree kernel, supersense kernel, bag of synonyms and hypernyms kernel) linearly combined and weighted under different conditions. The experiments were carried out using support vector machines (Vapnik, 1998) as classifier.

We present results on the Classification of Semantic Relations between Nominals task at SemEval-2007, in which sentences containing ordered pairs of marked nominals, possibly semantically related, have to be classified. On this task, we achieve an overall F_1 of 71.8% (B category evaluation), largely outperforming all the baselines.

[1] These results appear in a paper currently under revision.

Proceedings of the 4th International Workshop on Semantic Evaluations (SemEval-2007), pages 141–144,
Prague, June 2007.

2 Kernel Methods for Relation Extraction

In order to implement the approach based on syntactic and semantic information, we employed a linear weighted combination of kernels, using support vector machines as classifier. We designed two families of basic kernels: syntactic kernels and semantic kernels. These basic kernels are combined by exploiting the closure properties of kernels. We define our composite kernel $K_C(x_1, x_2)$ as follows

$$\sum_{i=1}^{n} w_i \frac{K_i(x_1, x_2)}{\sqrt{K_i(x_1, x_1)K_i(x_2, x_2)}}, \quad (1)$$

where each basic kernel K_i is normalized and $w_i \in \{0, 1\}$ is the kernel weight. The normalization factor plays an important role in allowing us to integrate information from heterogeneous knowledge sources.

All basic kernels, but the tree kernel (see Section 2.1.3), are explicitly calculated as follows

$$K_i(x_1, x_2) = \langle \phi(x_1), \phi(x_2) \rangle, \quad (2)$$

where $\phi(\cdot)$ is the embedding vector. Even though the resulting feature space has high dimensionality, an efficient computation of Equation 2 can be carried out explicitly since the input representations defined below are extremely sparse.

2.1 Syntactic Kernels

Syntactic kernels are defined over the whole sentence where the candidate nominals appear.

2.1.1 Global Context Kernel

Bunescu and Mooney (2005) and Giuliano et al. (2006) successfully exploited the fact that relations between named entities are generally expressed using only words that appear simultaneously in one of the following three contexts.

Fore-Between Tokens before and between the two entities, e.g. *"the head of* $[ORG]$*, Dr.* $[PER]$*"*.

Between Only tokens between the two entities, e.g. *"*$[ORG]$ *spokesman* $[PER]$*"*.

Between-After Tokens between and after the two entities, e.g. *"*$[PER]$*, a* $[ORG]$ *professor"*.

Here, we investigate whether this assumption is also correct for semantic relations between nominals. Our global context kernel operates on the contexts defined above, where each context is represented using a *bag-of-words*. More formally, given a relation example R, we represent a context C as a row vector

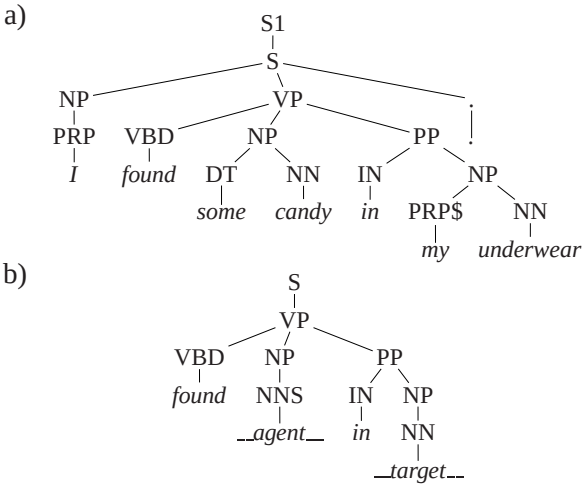

Figure 1: A *content-container* relation test sentence parse tree (a) and the corresponding RT structure (b).

$$\phi_C(R) = (tf(t_1, C), tf(t_2, C), \ldots, tf(t_l, C)) \in \mathbb{R}^l, \quad (3)$$

where the function $tf(t_i, C)$ records how many times a particular token t_i is used in C. Note that this approach differs from the standard bag-of-words as punctuation and stop words are included in ϕ_C, while the nominals are not. To improve the classification performance, we have further extended ϕ_C to embed n-grams of (contiguous) tokens (up to $n = 3$). By substituting ϕ_C into Equation 2, we obtain the n-gram kernel K_n, which counts uni-grams, bi-grams, ..., n-grams that two patterns have in common[2]. The *Global Context* kernel $K_{GC}(R_1, R_2)$ is then defined as

$$K_{FB}(R_1, R_2) + K_B(R_1, R_2) + K_{BA}(R_1, R_2), \quad (4)$$

where K_{FB}, K_B and K_{BA} are n-gram kernels that operate on the Fore-Between, Between and Between-After patterns respectively.

2.1.2 Bag-of-Words Kernel

The bag-of-words kernel is defined as the previous kernel but it operates on the whole sentence.

2.1.3 Tree Kernel

Tree kernels can trigger automatic feature selection and represent a viable alternative to the man-

[2] In the literature, it is also called *n-spectrum* kernel.

ual design of attribute-value syntactic features (Moschitti, 2004). A tree kernel $K_T(t_1, t_2)$ evaluates the similarity between two trees t_1 and t_2 in terms of the number of fragments they have in common. Let N_t be the set of nodes of a tree t and $\mathcal{F} = \{f_1, f_2, \ldots, f_{|\mathcal{F}|}\}$ be the fragment space of t_1 and t_2. Then

$$K_T(t_1, t_2) = \sum_{n_i \in N_{t_1}} \sum_{n_j \in N_{t_2}} \Delta(n_i, n_j) \ , \quad (5)$$

where $\Delta(n_i, n_j) = \sum_{k=1}^{|\mathcal{F}|} I_k(n_i) \times I_K(n_j)$ and $I_k(n) = 1$ if k is rooted in n, 0 otherwise.

For this task, we defined an *ad-hoc* class of structured features (Moschitti et al., 2006), the Reduced Tree (RT), which can be derived from a sentence parse tree t by the following steps: (1) remove all the terminal nodes but those labeled as relation entities and those POS tagged as verbs, auxiliaries, prepositions, modals or adverbs; (2) remove all the internal nodes not covering any remaining terminal; (3) replace the entity words with placeholders that indicate the direction in which the relation should hold. Figure 1 shows a parse tree and the resulting RT structure.

2.2 Semantic Kernels

In (Giuliano et al., 2006), we used the local context kernel to infer semantic information on the candidate entities (i.e., roles played by the entities). As the task organizers provide the WordNet sense and role for each nominal, we directly use this information to enrich the feature space and do not include the local context kernel in the combination.

2.2.1 Bag of Synonyms and Hypernyms Kernel

By using the WordNet sense key provided, each nominal is represented by the bag of its synonyms and hypernyms (direct and inherited hypernyms). Formally, given a relation example R, each nominal N is represented as a row vector

$$\phi_N(R) = (f(t_1, N), f(t_2, N), \ldots, f(t_l, N)) \in \mathbb{R}^l, \quad (6)$$

where the binary function $f(t_i, N)$ records if a particular lemma t_i is contained into the bag of synonyms and hypernyms of N. The *bag of synonyms and hypernyms* kernel $K_{S\&H}(R_1, R_2)$ is defined as

$$K_{target}(R_1, R_2) + K_{agent}(R_1, R_2), \quad (7)$$

where K_{target} and K_{agent} are defined by substituting the embedding of the target and agent nominals into Equation 2 respectively.

2.2.2 Supersense Kernel

WordNet synsets are organized into 45 lexicographer files, based on syntactic category and logical groupings. E.g., *noun.artifact* is for nouns denoting man-made objects, *noun.attribute* for nouns denoting attributes for people and objects etc. The *supersense* kernel $K_{SS}(R_1, R_2)$ is a variant of the previous kernel that uses the names of the lexicographer files (i.e., the supersense) to index the feature space.

3 Experimental Setup and Results

Sentences have been tokenized, lemmatized, and POS tagged with TextPro[3]. We considered each relation as a different binary classification task, and each sentence in the data set is a positive or negative example for the relation. The direction of the relation is considered labelling the first argument of the relation as agent and the second as target.

All the experiments were performed using the SVM package SVMLight-TK[4], customized to embed our own kernels. We optimized the linear combination weights w_i and regularization parameter c using 10-fold cross-validation on the training set. We set the cost-factor j to be the ratio between the number of negative and positive examples.

Table 1 shows the performance on the test set. We achieve an overall F_1 of 71.8% (B category evaluation), largely outperforming all the baselines, ranging from 48.5% to 57.0%. The average training plus test running time for a relation is about 10 seconds on a Intel Pentium M755 2.0 GHz. Figure 2 shows the learning curves on the test set. For all relations but *theme-tool*, accurate classifiers can be learned using a small fraction of training.

4 Discussion and Conclusion

Experimental results show that our kernel-based approach is appropriate also to detect semantic relations between nominals. However, differently from relation extraction between named entities, there is not a common kernel setup for all relations. E.g.,

[3] http://tcc.itc.it/projects/textpro/

[4] http://ai-nlp.info.uniroma2.it/moschitti/

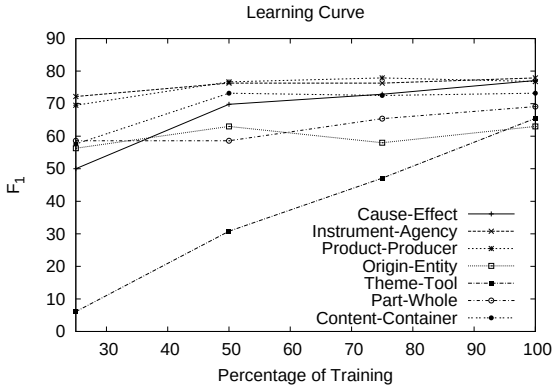

Figure 2: Learning curves on the test set.

Relation	P	R	F_1	Acc
Cause-Effect	67.3	90.2	77.1	72.5
Instrument-Agency	76.9	78.9	77.9	78.2
Product-Producer	76.2	77.4	76.8	68.8
Origin-Entity	62.2	63.9	63.0	66.7
Theme-Tool	69.2	62.1	65.5	73.2
Part-Whole	65.5	73.1	69.1	76.4
Content-Container	78.8	68.4	73.2	74.3
Avg	70.9	73.4	71.8	72.9

Table 1: Results on the test set.

for *content-container* we obtain the best performance combining the tree kernel and the bag of synonyms and hypernyms kernel; on the other hand, for *instrument-agency* the best performance is obtained by combining the global kernel and the supersense kernel. Surprisingly, the supersense kernel alone works quite well and obtains results comparable to the bag of synonyms and hypernyms kernel. This result is particularly interesting as a supersense tagger can easily provide a satisfactory accuracy (Ciaramita and Altun, 2006). On the other hand, obtaining an acceptable accuracy in word sense disambiguation (required for a realistic application of the bag of synonyms and hypernyms kernel) is impractical as a sufficient amount of training for at least all nouns is currently not available. Hence, the supersense could play a crucial role to improve the performance when approaching this task without the nominals disambiguated. To model the global context using the Fore-Between, Between and Between-After contexts did not produce a significant improvement with respect to the bag-of-words model. This is mainly due to the fact that examples have been collected from the Web using heuristic patterns/queries, most of which implying Between patterns/contexts (e.g., for the *cause-effect* relation "* comes from *", "* out of *" etc.).

5 Acknowledgements

Claudio Giuliano, Alberto Lavelli and Lorenza Romano are supported by the X-Media project (`http://www.x-media-project.org`), sponsored by the European Commission as part of the Information Society Technologies (IST) programme under EC grant number IST-FP6-026978.

References

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

Eugene Charniak. 2000. A maximum-entropy-inspired parser. In *Proceedings of the First Meeting of the North American Chapter of the Association for Computational Linguistics*, pages 132–139, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Massimiliano Ciaramita and Yasemin Altun. 2006. Broad-coverage sense disambiguation and information extraction with a supersense sequence tagger. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pages 594–602, Sydney, Australia, July.

Claudio Giuliano, Alberto Lavelli, and Lorenza Romano. 2006. Exploiting shallow linguistic information for relation extraction from biomedical literature. In *Proceedings of the Eleventh Conference of the European Chapter of the Association for Computational Linguistics (EACL-2006)*, Trento, Italy, 5-7 April.

Alessandro Moschitti, Daniele Pighin, and Roberto Basili. 2006. Semantic role labeling via tree kernel joint inference. In *Proceedings of the Tenth Conference on Computational Natural Language Learning, CoNLL-X*.

Alessandro Moschitti. 2004. A study on convolution kernels for shallow statistic parsing. In *Proceedings of the 42nd Meeting of the Association for Computational Linguistics (ACL'04), Main Volume*, pages 335–342, Barcelona, Spain, July.

Vladimir Vapnik. 1998. *Statistical Learning Theory*. John Wiley and Sons, New York, NY.