Evaluation of Local Features for Argument Identification and Classification for Semantic Role Labelling

Yvonne Samuelsson
Dept. of Linguistics
Stockholm University
yvonne.samuelsson@ling.su.se

Oscar Täckström
Dept. of Linguistics and Philology
SICS / Uppsala University
oscar@sics.se

Sumithra Velupillai
Dept. of Computer and Systems Sciences
Stockholm University / KTH
sumithra@dsv.su.se

1 Introduction

In recent years, automatic annotation of semantic roles in text, Semantic Role Labelling (SRL), has gained much interest and research attention. Semantic roles are useful in tasks such as question-answering, document summarization and information extraction.

For the CoNLL 2008 shared task on joint learning of syntactic and semantic dependencies (Surdeanu et al., 2008), we developed a pipelined SRL system (Samuelsson et al., 2008). In line with several previous SRL systems, predicate identification, argument identification and argument classification were separated into different learning tasks.

An important aspect of designing an SRL system is to identify which features are useful. Many features have been designed and used throughout the years. Xue and Palmer (2004) showed that different features are useful for different tasks. However, to our knowledge, no systematic, comprehensive evaluation of these features has been performed.

This paper presents a first step towards such an evaluation for the sub-tasks of argument identification and classification.

2 Features

For the feature evaluation, we implemented those features commonly used in systems developed from Gildea and Jurafsky (2002) and onward. We analyse the impact of features belonging to different feature types. By a *feature type* we mean e.g. DEPREL,[1] while by a *feature* we mean e.g. DEPREL=SUBJ. In total, over 100 feature types are examined in the evaluation.

The classification framework in this paper is local, in the sense that in each classification decision, only one predicate and one (candidate) argument is taken into account. We further assume that the classifier to be learned is parametrized by a weight vector, and that data instances are represented as vectors of features.

We distinguish between *single node* and *node pair* feature types, where the former represents atomic nodes in a dependency graph, and the latter represents the relation between a pair of nodes. While some feature types are defined on the token level, e.g. (PoS/LEMMA), most of the feature types are defined on dependency graphs.

All single node feature types defined on the dependency level have been applied to four different nodes: the current node itself, the parent of the current node, and the left and right sibling of the current node. They can be applied to both the predicate node and the (candidate) argument node.

- Form, Lemma, PoS-tag, Deprel, Verb voice, and Form initial case
- Children + self number of words, Form seq and BoW, Lemma seq and BoW, PoS-tag seq and BoW
- Children Deprel seq and BoW
- Children + self content Form seq and BoW, Lemma seq and BoW, PoS-tag seq and BoW
- Children content Deprel seq and BoW
- Immediate children Form seq and BoW, Lemma seq and BoW, PoS-tag seq and BoW, Deprel seq and BoW
- First child Form, Lemma, PoS-tag, Deprel
- Last child Form, Lemma, PoS-tag, Deprel

Additionally, a number of node pair features have been evaluated, where the pair consists of the predicate and the (candidate) argument node.

- Relative Position (before/after), Distance in Words
- Full path (all deprels), Pos full path (starting with the PoS-tag of the argument node, ending with the PoS-tag of the predicate node)
- Mid path (lowest common node), Pos mid path
- Short path (first and last deprel only), Pos short path
- Full distance in deprels, Mid distance in deprels

[1] Short for *Dependency Relation*

3 Evaluation

In order to evaluate which features are most useful for argument identification and classification, we apply a feature set selection technique. We use the SVM-RFE (Support Vector Machine - Recursive Feature Elimination) algorithm (Guyon et al., 2002), which makes use of the duality between the feature space and the instance space in weight vector parametrized models. More specifically, we have used SVM-RFE(OVA) proposed by Zhou and Tuck (2007), which is designed for multi-class problems. The RFE conducts feature selection in a backward elimination procedure, at each iteration removing the features which least influence the decision boundary. The feature selection is applied to features, not feature types. However, by selecting the top performing features we can also extract the top performing feature types. The LIBLINEAR software (Fan et al., 2008) was used in the implementation of SVM-RFE.

4 Preliminary Results

We have performed our experiments on a subset of the Wall Street Journal. Preliminary results on the argument classification task show that the microaverage F_1-score is generally improved when the number of features is reduced. For example, for the classification task, less than 10k features are optimal.

In the classification task, when looking at the F_1-score per argument label, arguments of verbal predicates get a minor improvement when reducing the number of features to 10k. This shows that features can be removed to improve system speed, without loss of accuracy.

However, for arguments of nominal predicates, we also get a major improvement in the F_1-score per argument label, when reducing the number of features to 10k. These features should thus be removed to improve both speed and accuracy. Moreover, different argument types may have different cut-off points in terms of the number of features removed. We see for instance that the F_1-scores for some non-core arguments verbal predicates, do not have the same cut-off point as core aruments.

The microaverage F_1-score for arguments of verbal predicates is generally much higher than for arguments of nominal predicates. Generally argument structures for nominal predicates are more difficult to learn, and benefit from different feature types. For example, argument nodes and lexical features seem to be more important for nominal predicates.

In future work we will evaluate both the classification and the identification task in more detail and on a larger corpus. We need to find the cut-off points for both learning tasks, as well as for nominal and verbal predicates. The specific features and feature types to be removed need to be further analysed.

Acknowledgments

We thank our supervisors in the Machine Learning 2 course at GSLT, Joakim Nivre and Björn Gambäck, for advice and support. Computations were performed on UPPMAX (project p2005028).

References

Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. 2008. LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, 9:1871–1874.

Daniel Gildea and Daniel Jurafsky. 2002. Automatic labeling of semantic roles. *Computational Linguistics*, 28(3):245–288.

Isabelle Guyon, Jason Weston, Stephen Barnhill, and Vladimir Vapnik. 2002. Gene selection for cancer classification using support vector machines. *Machine Learning*, 46(1-3):389–422.

Yvonne Samuelsson, Oscar Täckström, Sumithra Velupillai, Johan Eklund, Mark Fishel, and Markus Saers. 2008. Mixing and blending syntactic and semantic dependencies. In *CoNLL 2008: Proceedings of the Twelfth Conference on Computational Natural Language Learning*, pages 248–252, Manchester, England, August. Coling 2008 Organizing Committee.

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, Great Britain.

Nianwen Xue and Martha Palmer. 2004. Calibrating features for semantic role labeling. In Dekang Lin and Dekai Wu, editors, *Proceedings of EMNLP 2004*, pages 88–94, Barcelona, Spain, July. Association for Computational Linguistics.

Xin Zhou and David P. Tuck. 2007. MSVM-RFE: extensions of SVM-RFE for multiclass gene selection on DNA microarray data. *Bioinformatics*, 23(9):1106–1114.