# Improving Semantic Role Classification with Selectional Preferences

**Beñat Zapirain, Eneko Agirre**
IXA NLP Group
Basque Country Univ.
{benat.zapirain,e.agirre}@ehu.es

**Lluís Màrquez**
TALP Research Center
Technical Univ. of Catalonia
lluism@lsi.upc.edu

**Mihai Surdeanu**
Stanford NLP Group
Stanford Univ.
mihais@stanford.edu

## Abstract

This work incorporates Selectional Preferences (SP) into a Semantic Role (SR) Classification system. We learn separate selectional preferences for noun phrases and prepositional phrases and we integrate them in a state-of-the-art SR classification system both in the form of features and individual class predictors. We show that the inclusion of the refined SPs yields statistically significant improvements on both in domain and out of domain data (14.07% and 11.67% error reduction, respectively). The key factor for success is the combination of several SP methods with the original classification model using meta-classification.

## 1 Introduction

Semantic Role Labeling (SRL) is the process of extracting simple event structures, i.e., "who" did "what" to "whom", "when" and "where". Current systems usually perform SRL in two pipelined steps: argument *identification* and argument *classification*. While identification is mostly syntactic, classification requires semantic knowledge to be taken into account. Semantic information is usually captured through lexicalized features on the predicate and the head–word of the argument to be classified. Since lexical features tend to be sparse, SRL systems are prone to overfit the training data and generalize poorly to new corpora.

Indeed, the SRL evaluation exercises at CoNLL-2004 and 2005 (Carreras and Màrquez, 2005) observed that all systems showed a significant performance degradation ($\sim$10 $F_1$ points) when applied to test data from a different genre of that of the training set. Pradhan et al. (2008) showed that this performance degradation is essentially caused by the argument classification subtask, and suggested the lexical data sparseness as one of the main reasons. The same authors studied the contribution of the different feature types in SRL and concluded that the lexical features were the most salient features in argument classification (Pradhan et al., 2007).

In recent work, we showed (Zapirain et al., 2009) how automatically generated selectional preferences (SP) for verbs were able to perform better than pure lexical features in a role classification experiment, disconnected from a full-fledged SRL system. SPs introduce semantic generalizations on the type of arguments preferred by the predicates and, thus, they are expected to improve results on infrequent and unknown words. The positive effect was especially relevant for out-of-domain data. In this paper we advance (Zapirain et al., 2009) in two directions:
(1) We learn separate SPs for prepositions and verbs, showing improvement over using SPs for verbs alone.
(2) We integrate the information of several SP models in a state-of-the-art SRL system (SwiRL[1]) and show significant improvements in SR classification. The key for the improvement lies in a meta-classifier, trained to select among the predictions provided by several role classification models.

## 2 SPs for SR Classification

SPs have been widely believed to be an important knowledge source when parsing and performing SRL, especially role classification. Still, present parsers and SRL systems use just lexical features, which can be seen as the most simple form of SP,

[1] http://www.surdeanu.name/mihai/swirl/

where the headword needs to be seen in the training data, and otherwise the SP is not satisfied. Gildea and Jurafsky (2002) showed barely significant improvements in semantic role classification of NPs for FrameNet roles using distributional clusters. In (Erk, 2007) a number of SP models are tested in a pseudo-task related to SRL. More recently, we showed (Zapirain et al., 2009) that several methods to automatically generate SPs generalize well and outperform lexical match in a large dataset for semantic role classification, but the impact on a full system was not explored.

In this work we apply a subset of the SP methods proposed in (Zapirain et al., 2009). These methods can be split in two main families, depending on the resource used to compute similarity: WordNet-based methods and distributional methods. Both families define a similarity score between a word (the headword of the argument to be classified) and a set of words (the headwords of arguments of a given role).

**WordNet-based similarity**: One of the models that we used is based on Resnik's similarity measure (1993), referring to it as $res$. The other model is an in-house method (Zapirain et al., 2009), referred as $wn$, which only takes into account the depth of the most common ancestor, and returns SPs that are as specific as possible.

**Distributional similarity**: Following (Zapirain et al., 2009) we considered both first order and second order similarity. In first order similarity, the similarity of two words was computed using the cosine (or Jaccard measure) of the co-occurrence vectors of the two words. Co-occurrence vectors where constructed using freely available software (Padó and Lapata, 2007) run over the British National Corpus. We used the optimal parameters (Padó and Lapata, 2007, p. 179). We will refer to these similarities as $sim_{cos}$ and $sim_{Jac}$, respectively. In contrast, second order similarity uses vectors of similar words, i.e., the similarity of two words was computed using the cosine (or Jaccard measure) between the thesaurus entries of those words in Lin's thesaurus (Lin, 1998). We refer to these as $sim^2_{cos}$ and $sim^2_{Jac}$.

Given a target sentence with a verb and its arguments, the task of SR classification is to assign the correct role to each of the arguments. When using SPs alone, we only use the headwords of the arguments, and each argument is classified independently of the rest. For each headword, we select the role ($r$) of the verb ($c$) which fits best the head word ($w$), where the goodness of fit ($SP_{sim}(v, r, w)$) is modeled using one of the similarity models above, between the headword $w$ and the headwords seen in training data for role $r$ of verb $v$. This selection rule is formalized as follows:

$$R_{sim}(v, w) = \arg \max_{r \in Roles(v)} SP_{sim}(v, r, w) \quad (1)$$

In our previous work (Zapirain et al., 2009), we modelled SPs for pairs of predicates (verbs) and arguments, independently of the fact that the argument is a core argument (typically a noun) or an adjunct argument (typically a prepositional phrase). In contrast, (Litkowski and Hargraves, 2005) show that prepositions have SPs of their own, especially when functioning as adjuncts. We therefore decided to split SPs according to whether the potential argument is a Prepositional Phrase (PP) or a Noun Phrase (NP). For NPs, which tend to be core arguments[2], we use the SPs of the verb (as formalized above). For PPs, which have an even distribution between core and adjunct arguments, we use the SPs of the prepositions alone, ignoring the verbs. Implementation wise, this means that in Eq. (1), we change $v$ for $p$, where $p$ is the preposition heading the PP.

## 3 Experiments with SPs in isolation

In this section we evaluate the use of SPs for classification in isolation, i.e., we use formula 1, and no other information. In addition we contrast the use of both verb-role and preposition-role SPs, as compared to the use of verb-role SPs alone.

The dataset used in these experiments (and in Section 4) is the same as provided by the CoNLL-2005 shared task on SRL (Carreras and Màrquez, 2005). This dataset comprises several sections of the PropBank corpus (news from the WSJ) as well as an extract of the Brown Corpus. Sections 02-21 are used for generating the SPs and training, Section 00 for development, and Section 23 for testing, as customary. The Brown Corpus is used for out-of-domain testing, but due to the limited size of the provided section, we extended it with instances from SemLink[3]. Since the focus of this work is on argument

[2] In our training data, NPs are adjuncts only 5% of the times

[3] http://verbs.colorado.edu/semlink/

| | Verb-Role SPs | | | | | | Preposition-Role and Verb-Role SPs | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | WSJ-test | | | Brown | | | WSJ-test | | | Brown | | |
| | prec. | rec. | $F_1$ | prec. | rec. | $F_1$ | prec. | rec. | $F_1$ | prec. | rec. | $F_1$ |
| lexical | **70.75** | 26.66 | 39.43 | **59.39** | 05.51 | 10.08 | **82.98** | 43.77 | 57.31 | **68.47** | 13.60 | 22.69 |
| $SP_{res}$ | 45.07 | 37.11 | 40.71 | 36.34 | 27.58 | 31.33 | 63.47 | 53.24 | 57.91 | 55.12 | 44.15 | 49.03 |
| $SP_{wn}$ | 55.44 | 45.58 | 50.03 | 41.76 | 31.58 | 35.96 | 65.70 | 63.88 | 64.78 | 60.08 | 48.10 | 53.43 |
| $SP_{sim_{Jac}}$ | 48.85 | 46.38 | 47.58 | 42.10 | 34.34 | 37.82 | 61.83 | 61.40 | 61.61 | 55.42 | 53.45 | 54.42 |
| $SP_{sim_{cos}}$ | 53.13 | 50.44 | 51.75 | 43.24 | 35.27 | 38.85 | 64.67 | 64.22 | 64.44 | 56.56 | 54.54 | 55.53 |
| $SP_{sim^2_{Jac}}$ | 61.76 | **58.63** | **60.16** | 51.97 | **42.39** | **46.69** | 70.82 | **70.33** | **70.57** | 62.37 | **60.15** | **61.24** |
| $SP_{sim^2_{cos}}$ | 61.12 | 58.12 | 59.63 | 51.92 | 42.35 | 46.65 | 70.28 | 69.80 | 70.04 | 62.36 | 60.14 | 61.23 |

Table 1: Results for SPs in isolation, left for verb SPs, and right both preposition and verb SPs.

| Labels proposed by the base models |
|---|
| Number of base models that proposed this datum's label |
| List of actual base models that proposed this datum's label |

Table 2: Features of the binary meta-classifier.

classification, we use the gold PropBank data to identify argument boundaries. Considering that SPs can handle only nominal arguments, in these experiments we used only arguments mapped to NPs and PPs containing a nominal head. From the training sections, we extracted over 140K such arguments for the supervised generation of SPs. The development and test sections contain over 5K and 8K examples, respectively, and the portion of the Brown Corpus comprises an amount of 8.1K examples.

Table 1 lists the results of the different SPs in isolation. The results reported in the left part of Table 1 are comparable to those we reported in (Zapirain et al., 2009). The differences are due to the fact that we do not discard roles like MOD, DIS, NEG and that our previous work used only the subset of the data that could be mapped to VerbNet (around 50%). All in all, the table shows that splitting SPs into verb and preposition SPs yields better results, both in precision and recall, improving $F_1$ up to 10 points in some cases.

## 4 Integrating SPs in a SRL system

For these experiments we modified SwiRL (Surdeanu et al., 2007): (a) we matched the gold boundaries against syntactic constituents predicted internally using the Charniak parser (Charniak, 2000); and (b) we classified these constituents with their semantic role using a modified version of SwiRL's feature set.

We explored two different strategies for integrating SPs in SwiRL. The first, obvious method is to extend SwiRL's feature set with features that model the preferences of the SPs, i.e., for each SP model $SP_i$ we add a feature whose value is $R_i$. The second method combines SwiRL's classification model and our SP models using meta-classification. We opted for a binary classification approach: first, for each constituent we generate $n$ datums, one for each distinct role label proposed by the pool of base models; then we use a binary meta-classifier to label each candidate role as correct or incorrect. Table 2 lists the features of the meta-classifier. We trained the meta-classifier on the usual PropBank training partition, using cross-validation to generate outputs for the base models that require the same training material. At prediction time, for each candidate constituent we selected the role label that was classified as correct with the highest confidence.

Table 3 compares the performance of both combination approaches against the standalone SwiRL classifier. We show results for both core arguments (Core), adjunct arguments (Arg) and all arguments combined (All). In the table, the SwiRL+$SP_*$ models stand for SwiRL classifiers enhanced with one feature from the corresponding SP. Adding more than one SP-based feature to SwiRL did not improve results. Our conjecture is that the SwiRL classifier enhanced with SP-based features does not learn relevant weights for these features because their signal is "drowned" by SwiRL's large initial feature set and the correlation between the different SPs. This observation motivated the development of the meta-classifier. The meta-classifier shown in the table combines the output of the SwiRL+$SP_*$ models with the predictions of SP models used in isolation. We implemented the meta-classifier using Support Vector Machines (SVM)[4] with a quadratic polynomial kernel, and

[4] http://svmlight.joachims.org

| | WSJ-test | | | Brown | | |
|---|---|---|---|---|---|---|
| | Core | Adj | All | Core | Adj | All |
| SwiRL | 93.25 | 81.31 | 90.83 | 84.42 | 57.76 | 79.52 |
| +$SP_{Res}$ | 93.17 | 81.08 | 90.76 | 84.52 | 59.24 | 79.86 |
| +$SP_{wn}$ | 92.88 | 81.11 | 90.56 | 84.26 | 59.69 | 79.73 |
| +$SP_{sim_{Jac}}$ | 93.37 | 80.30 | 90.86 | 84.43 | 59.54 | 79.83 |
| +$SP_{sim_{cos}}$ | 93.33 | 80.92 | 90.87 | 85.14 | 60.16 | 80.50 |
| +$SP_{sim^2_{Jac}}$ | 93.03 | 82.75 | 90.95 | 85.62 | 59.63 | 80.75 |
| +$SP_{sim^2_{cos}}$ | 93.78 | 80.56 | 91.23 | 84.95 | 61.01 | 80.48 |
| Meta | **94.37** | **83.40** | **92.12** | **86.20** | **63.40** | **81.91** |

Table 3: Classification accuracy for the combination approaches. +$SP_x$ stands for SwiRL plus each SP model.

$C = 0.01$ (tuned in development).

Table 3 indicates that four out of the six SwiRL+$SP_*$ models perform better than SwiRL in domain (WSJ-test), and all of them outperform SwiRL out of domain (Brown). However, the improvements are small and, generally, not statistically significant. On the other hand, the meta-classifier outperforms SwiRL both in domain (14.07% error reduction) and out of domain (11.67% error reduction), and the differences are statistically significant (measured using two-tailed paired t-test at 99% confidence interval on 100 samples generated using bootstrap resampling). We also implemented two unsupervised voting baselines, one unweighted (each base model has the same weight) and one weighted (each base model is weighted by its accuracy in development). However, none of these baselines outperformed the standalone SwiRL classifier. This is further proof that, for SR classification, meta-classification is crucial because it can learn the distinct specializations of the various base models.

Finally, Table 3 shows that our approach yields consistent improvements for both core and adjunct arguments. Out of domain, we see a bigger accuracy improvement for adjunct arguments (5.64 absolute points) vs. core arguments (1.78 points). This is to be expected, as most core arguments fall under the Arg0 and Arg1 classes, which can typically be disambiguated based on syntactic information, i.e., subject vs. object. On the other hand, there are no syntactic hints for adjunct arguments, so the system learns to rely more on SP information in this case.

## 5 Conclusions

This paper is the first work to show that SPs improve a state-of-the-art SR classification system. Several decisions were crucial for success: (a) we deployed separate SP models for verbs and prepositions, which in conjunction outperform SP models for verbs alone; (b) we incorporated SPs into SR classification using a meta-classification approach that combines eight base models, developed from variants of a state-of-the-art SRL system and the above SP models. We show that the resulting system outperforms the original SR classification system for arguments mapped to nominal or prepositional constituents. The improvements are statistically significant both on in-domain and out-of-domain data sets.

## Acknowledgments

This work was partially supported by projects KNOW-2 (TIN2009-14715-C04-01 / 04), KYOTO (ICT-2007-211423) and OpenMT-2 (TIN2009-14675C03)

## References

X. Carreras and L. Màrquez. 2005. Introduction to the CoNLL-2005 Shared Task: Semantic role labeling. In *Proc. of CoNLL*.

E. Charniak. 2000. A maximum-entropy-inspired parser. In *Proc. of NAACL*.

K. Erk. 2007. A simple, similarity-based model for selectional preferences. In *Proc. of ACL*.

D. Gildea and D. Jurafsky. 2002. Automatic labeling of semantic roles. *Computational Linguistics*, 28(3).

D. Lin. 1998. Automatic retrieval and clustering of similar words. In *Proc. of COLING-ACL*.

K. Litkowski and O. Hargraves. 2005. The preposition project. In *Proceedings of the Workshop on The Linguistic Dimensions of Prepositions and their Use in Computational Linguistic Formalisms and Applications*.

S. Padó and M. Lapata. 2007. Dependency-based construction of semantic space models. *Computational Linguistics*, 33(2).

S. Pradhan, W. Ward, and J. Martin. 2007. Towards robust semantic role labeling. In *Proc. of NAACL-HLT*.

S. Pradhan, W. Ward, and J. Martin. 2008. Towards robust semantic role labeling. *Computational Linguistics*, 34(2).

P. Resnik. 1993. Semantic classes and syntactic ambiguity. In *Proc. of HLT*.

M. Surdeanu, L. Màrquez, X. Carreras, and P.R. Comas. 2007. Combination strategies for semantic role labeling. *Journal of Artificial Intelligence Research*, 29.

B. Zapirain, E. Agirre, and L. Màrquez. 2009. Generalizing over lexical features: Selectional preferences for semantic role classification. In *Proc. of ACL-IJCNLP*.