Selectional Preferences for Semantic Role Classification

Beñat Zapirain*
University of the Basque Country

Eneko Agirre**
University of the Basque Country

Lluís Màrquez†
Universitat Politècnica de Catalunya

Mihai Surdeanu‡
University of Arizona

This paper focuses on a well-known open issue in Semantic Role Classification (SRC) research: the limited influence and sparseness of lexical features. We mitigate this problem using models that integrate automatically learned selectional preferences (SP). We explore a range of models based on WordNet and distributional-similarity SPs. Furthermore, we demonstrate that the SRC task is better modeled by SP models centered on both verbs and prepositions, rather than verbs alone. Our experiments with SP-based models in isolation indicate that they outperform a lexical baseline with 20 F_1 points in domain and almost 40 F_1 points out of domain. Furthermore, we show that a state-of-the-art SRC system extended with features based on selectional preferences performs significantly better, both in domain (17% error reduction) and out of domain (13% error reduction). Finally, we show that in an end-to-end semantic role labeling system we obtain small but statistically significant improvements, even though our modified SRC model affects only approximately 4% of the argument candidates. Our post hoc error analysis indicates that the SP-based features help mostly in situations where syntactic information is either incorrect or insufficient to disambiguate the correct role.

* Informatika Fakultatea, Manuel Lardizabal 1, 20018 Donostia, Basque Country.
E-mail: benat.zapirain@ehu.es.
** Informatika Fakultatea, Manuel Lardizabal 1, 20018 Donostia, Basque Country.
E-mail: e.agirre@ehu.es.
† UPC Campus Nord (Omega building), Jordi Girona 1–3, 08034 Barcelona, Catalonia.
E-mail: lluism@lsi.upc.edu.
‡ 1040 E. 4th Street, Tucson, AZ 85721. E-mail: msurdeanu@arizona.edu.

Submission received: 14 November 2011; revised submission received: 31 May 2012; accepted for publication: 15 August 2012.

doi:10.1162/COLI_a_00145

1. Introduction

Semantic Role Labeling (SRL) is the problem of analyzing clause predicates in text by identifying arguments and tagging them with semantic labels indicating the role they play with respect to the predicate. Such sentence-level semantic analysis allows the determination of *who* did *what* to *whom*, *when* and *where*, and thus characterizes the participants and properties of the events established by the predicates. For instance, consider the following sentence, in which the arguments of the predicate *to_send* have been annotated with their respective semantic roles.[1]

(1) [Mr. Smith]$_{Agent}$ *sent* [the report]$_{Object}$ [to me]$_{Recipient}$ [this morning]$_{Temporal}$.

Recognizing these event structures has been shown to be important for a broad spectrum of NLP applications. Information extraction, summarization, question answering, machine translation, among others, can benefit from this shallow semantic analysis at sentence level, which opens the door for exploiting the semantic relations among arguments (Boas 2002; Surdeanu et al. 2003; Narayanan and Harabagiu 2004; Melli et al. 2005; Moschitti et al. 2007; Higashinaka and Isozaki 2008; Surdeanu, Ciaramita, and Zaragoza 2011). In Màrquez et al. (2008) the reader can find a broad introduction to SRL, covering several historical and definitional aspects of the problem, including also references to the main resources and systems.

State-of-the-art systems leverage existing hand-tagged corpora (Fillmore, Ruppenhofer, and Baker 2004; Palmer, Gildea, and Kingsbury 2005) to learn supervised machine learning systems, and typically perform SRL in two sequential steps: argument *identification* and argument *classification*. Whereas the former is mostly a syntactic recognition task, the latter usually requires semantic knowledge to be taken into account. The semantic knowledge that most current systems capture from text is basically limited to the predicates and the lexical units contained in their arguments, including the argument head. These "lexical features" tend to be sparse, especially when the training corpus is small, and thus SRL systems are prone to overfit the training data and generalize poorly to new corpora (Pradhan, Ward, and Martin 2008). As a simplified example of the effect of sparsity, consider the following sentences occurring in an imaginary training data set for SRL:

(2) [JFK]$_{Patient}$ *was_assassinated* [in Dallas]$_{Location}$

(3) [John Lennon]$_{Patient}$ *was_assassinated* [in New York]$_{Location}$

(4) [JFK]$_{Patient}$ *was_assassinated* [in November]$_{Temporal}$

(5) [John Lennon]$_{Patient}$ *was_assassinated* [in winter]$_{Temporal}$

All four sentences share the same syntactic structure, so the lexical features (i.e., the words *Dallas*, *New York*, *November*, and *winter*) represent the most relevant knowledge for discriminating between the *Location* and *Temporal* adjunct labels in learning.

1 For simplicity, in this paper we talk about arguments in the most general sense. Unless noted otherwise, **argument** will refer to both core-arguments (*Agent*, *Patient*, *Instrument*, etc.) and adjuncts (*Manner*, *Temporal*, *Location*, etc.).

The problem is that, as in the following sentences, for the same predicate, one may encounter similar expressions with new words like *Texas* or *December*, which the classifiers cannot match with the lexical features seen during training, and thus become useless for classification:

(6) [Smith] *was_assassinated* [in Texas]

(7) [Smith] *was_assassinated* [in December]

This problem is exacerbated when SRL systems are applied to texts coming from new domains where the number of new predicates and argument heads increases considerably. The CoNLL-2004 and 2005 evaluation exercises on semantic role labeling (Carreras and Màrquez 2004, 2005) reported a significant performance degradation of around 10 F_1 points when applied to out-of-domain texts from the Brown corpus. Pradhan, Ward, and Martin (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.

In this work, we will focus on Semantic Role Classification (SRC), and we will show that selectional preferences (SP) are useful for generalizing lexical features, helping fight sparseness and domain shifts, and improving SRC results. Selectional preferences try to model the kind of words that can fill a specific argument of a predicate, and have been widely used in computational linguistics since the early days (Wilks 1975). Both semantic classes from existing lexical resources like WordNet (Resnik 1993b) and distributional similarity based on corpora (Pantel and Lin 2000) have been successfully used for acquiring selectional preferences, and in this work we have used several of those models.

The contributions of this work to the field of SRL are the following:

1. We formalize and implement a method that applies several selectional preference models to Semantic Role Classification, introducing for the first time the use of selectional preferences for prepositions, in addition to selectional preferences for verbs.
2. We show that the selectional preference models are able to generalize lexical features and improve role classification performance in a controlled experiment disconnected from a complete SRL system. The positive effect is consistently observed in all variants of WordNet and distributional similarity measures and is especially relevant for out-of-domain data. The separate learning of SPs for verbs and prepositions contributes significantly to the improvement of the results.
3. We integrate the information of several SP models in a state-of-the-art SRL system (*SwiRL*)[2] and obtain significant improvements in semantic role classification and, as a consequence, in the end-to-end SRL task. The key for the improvement lies in the combination of the predictions provided by *SwiRL* and the several role classification models based on selectional preferences.

2 http://surdeanu.info/mihai/swirl/.

4. We present a manual analysis of the output of the combined role classification system. By observing a set of real examples, we categorized and quantified the situations in which SP models tend to help role classification. By inspecting also a set of negative cases, this analysis also sheds light on the limitations of the current approach and identifies opportunities for further improvements.

The use of selectional preferences for improving role classification was first presented in Zapirain, Agirre, and Màrquez (2009), and later extended in Zapirain et al. (2010) to a full-fledged SRC system. In the current paper, we provide more detailed background information and details of the selectional preference models, as well as complementary experiments on the integration in a full-fledged system. More importantly, we incorporate a detailed analysis of the output of the system, comparing it with that of a state-of-the-art SRC system not using SPs.

The rest of the paper is organized as follows. Section 2 provides background on the automatic acquisition of selectional preference, and its recent relation to the semantic role labeling problem. In Section 3, the SP models investigated in this paper are explained in all their variants. The results of the SP models in laboratory conditions are presented in Section 4. Section 5 describes the method for integrating the SP models in a state-of-the-art SRL system and discusses the results obtained. In Section 6 the qualitative analysis of the system output is presented, including a detailed discussion of several examples. Finally, Section 7 concludes and outlines some directions for future research.

2. Background

The simplest model for generating selectional preferences would be to collect all heads filling each role of the target predicate. This is akin to the lexical features used by current SRL systems, and we refer to this model as the **lexical model**. More concretely, the lexical model for **verb**-*role* selectional preferences consists of the list of words appearing as heads of the *role* arguments of the predicate **verb**. This model can be extracted automatically from the SRL training corpus using straightforward techniques. When using this model for role classification, it suffices to check whether the head word of the argument matches any of the words in the lexical model. The lexical model is the baseline for our other SP models, all of which build on that model.

In order to generalize the lexical model, semantic classes can be used. Although in principle any lexical resource listing semantic classes for nouns could be applied, most of the literature has focused on the use of WordNet (Resnik 1993b). In the **WordNet-based model**, the words occurring in the lexical model are projected over the semantic hierarchy of WordNet, and the semantic classes which represent best those words are selected. Given a new example, the SRC system has to check whether the new word matches any of those semantic classes. For instance, in example sentences (2)–(5), the semantic class `<time period>` covers both training examples for *Temporal* (i.e., *November* and *winter*), and `<geographical area>` covers the examples for *Location*. When test words *Texas* and *December* occur in Examples (6) and (7), the semantic classes to which they belong can be used to tag the first as *Location* and the second as *Temporal*.

As an alternative to the use of WordNet, one can also apply automatically acquired distributional similarity thesauri. Distributional similarity methods analyze the co-occurrence patterns of words and are able to capture, for instance, that *December* is more closely related to *November* than to *Dallas* (Grefenstette 1992). Distributional similarity is typically used on-line (i.e., given a pair of words, their similarity is computed on the go),

but, in order to speed up its use, it has also been used to produce off-line a full thesauri, storing, for every word, the weighted list of all outstanding similar words (Lin 1998). In the **Distributional similarity model**, when test item *Texas* in Example (6) is to be labeled, the higher similarity to *Dallas* and *New York*, in contrast to the lower similarity to *November* and *winter*, would be used to label the argument with the *Location* role.

The automatic acquisition of selectional preferences is a well-studied topic in NLP. Many methods using semantic classes and selectional preferences have been proposed and applied to a variety of syntactic–semantic ambiguity problems, including syntactic parsing (Hindle 1990; Resnik 1993b; Pantel and Lin 2000; Agirre, Baldwin, and Martinez 2008; Koo, Carreras, and Collins 2008; Agirre et al. 2011), word sense disambiguation (Resnik 1993a; Agirre and Martinez 2001; McCarthy and Carroll 2003), pronoun resolution (Bergsma, Lin, and Goebel 2008) and named-entity recognition (Ratinov and Roth 2009). In addition, selectional preferences have been shown to be effective to improve the quality of inference and information extraction rules (Pantel et al. 2007; Ritter, Mausam, and Etzioni 2010). In some cases, the aforementioned papers do not mention selectional preferences, but all of them use some notion of preferring certain semantic types over others in order to accomplish their respective task.

In fact, one could use different notions of semantic types. In one extreme, we would have a small set of coarse semantic classes. For instance, some authors have used the 26 so-called "semantic fields" used to classify all nouns in WordNet (Agirre, Baldwin, and Martinez 2008; Agirre et al. 2011). The classification could be more fine-grained, as defined by the WordNet hierarchy (Resnik 1993b; Agirre and Martinez 2001; McCarthy and Carroll 2003), and other lexical resources could be used as well. Other authors have used automatically induced hierarchical word classes, clustered according to occurrence information from corpora (Koo, Carreras, and Collins 2008; Ratinov and Roth 2009). On the other extreme, each word would be its own semantic class, as in the lexical model, but one could also model selectional preference using distributional similarity (Grefenstette 1992; Lin 1998; Pantel and Lin 2000; Erk 2007; Bergsma, Lin, and Goebel 2008). In this paper we will focus on WordNet-based models that use the whole hierarchy and on distributional similarity models, and we will use the lexical model as baseline.

2.1 WordNet-Based Models

Resnik (1993b) proposed the modeling of selectional preferences using semantic classes from WordNet and applied the model to tackle some ambiguity issues in syntax, such as noun-compounds, coordination, and prepositional phrase attachment. Given two alternative structures, Resnik used selectional preferences to choose the attachment maximizing the fitness of the head to the selectional preferences of the attachment points. This is similar to our task, but in our case we compare the target head to the selectional preference models for each possible role label (i.e., given a verb and the head of an argument, we need to find the role with the selectional preference that fits the head best).

In Resnik's model, he first characterizes the restrictiveness of the selectional preference of an argument position r of a governing predicate p, noted as $R(p,r)$. For that, given a set of classes C from the WordNet nominal hierarchies, he takes the relative entropy or Kullback-Leibler distance between the prior distribution $P(C)$ and the posterior distribution $P(C|p,r)$:

$$R(p,r) = \sum_{c \in C} P(c|p,r) log \frac{P(c|p,r)}{P(c)} \quad (1)$$

The priors can be computed from any corpora, computing frequencies of classes and using maximum likelihood estimates. The frequencies for classes cannot be directly observed, but they can be estimated from the lexical frequencies of the nouns under the class, as in Equation (2). Note that in WordNet, hypernyms ("hyp" for short) correspond to superclass relations, and therefore $hyp(n)$ returns all superclasses of noun n.

$$freq(c) = \sum_{\{n|c \in hyp(n)\}} freq(n) \tag{2}$$

A complication arises because of the polysemy of nouns. If each occurrence of a noun counted once in all classes that its senses belong to, polysemous nouns would account for more probability mass than monosemous nouns, even if they occurred the same number of times. As a solution, the frequency of polysemous nouns is split among its senses uniformly. For instance, the probability of the class `<time period>` can be estimated according to the frequencies of nouns like *November*, *spring*, and the rest of nouns under it. *November* has a single sense, so every occurrence counts as 1, but *spring* has six different senses, so each occurrence should only count as 0.16. Note that with this method we are implicitly dealing with the word sense ambiguity problem. When encountering a polysemous noun as an argument of a verb, we record the occurrence of all of its senses. Given enough occurrences of nouns, the classes generalizing the intended sense of the nouns will gather more counts than competing classes. In the example, `<time period>` would have 1.16 compared with 0.16 `<tool>` (i.e., for the metal elastic device meaning of *spring*). Researchers have used this fact to perform Word Sense Disambiguation using selectional preferences (Resnik 1993a; Agirre and Martinez 2001; McCarthy and Carroll 2003).

The posterior probability can be computed similarly, but it takes into account occurrences of the nouns in the required argument position of the predicate, and thus requires a corpus annotated with roles.

The selectional preference of a predicate p and role r for a head w_0 of any potential argument, noted as $SP_{Res}(p, r, w_0)$, is formulated as follows:[3]

$$SP_{Res}(p, r, w_0) = \max_{c_0 \in hyp(w_0)} \frac{P(c_0|p, r) log \frac{P(c_0|p,r)}{P(c_0)}}{R(p, r)} \tag{3}$$

The numerator formalizes the goodness of fit for the best semantic class c_0 that contains w_0. The hypernym (i.e., superclass) of w_0 yielding the maximum value is chosen. The denominator models how restrictive the selectional preference is for p and r, as modeled in Equation (1).

Variations of Resnik's idea to find a suitable level of generalization have been explored in later years. Li and Abe (1998) applied the minimum-description length principle. Alternatively, Clark and Weir (2002) devised a procedure to decide when a class should be preferred rather than its children.

Brockmann and Lapata (2003) compared several *class-based* models (including Resnik's selectional preferences) on a syntactic plausibility judgment task for German.

3 We slightly modified the notation of Resnik (1993b) in order to be coherent with the formulae presented in this paper.

The models return weights for (verb, syntactic_function, noun) triples, and correlation with human plausibility judgment is used for evaluation. Resnik's selectional preference scored best among WordNet-based methods (Li and Abe 1998; Clark and Weir 2002). Despite its earlier publication, Resnik's method is still the most popular representative among WordNet-based methods (Padó, Padó, and Erk 2007; Erk, Padó, and Padó 2010; Baroni and Lenci 2010). We also chose to use Resnik's model in this paper.

One of the disadvantages of the WordNet-based models, compared with the distributional similarity models, is that they require that the heads are present in WordNet. This limitation can negatively influence the coverage of the model, and also its generalization ability.

2.2 Distributional Similarity Models

Distributional similarity models assume that a word is characterized by the words it co-occurs with. In the simplest model, co-occurring words are taken from a fixed-size context window. Each word w would be represented by the set of words that co-occur with it, $T(w)$. In a more elaborate model, each word w would be represented as a vector of words $\vec{T}(w)$ with weights, where $\vec{T}_i(w)$ corresponds to the weight of the ith word in the vector. The weights can be calculated following a simple frequency of co-occurrence, or using some other formula.

Then, given two words w and w_0, their similarity can be computed using any similarity measure between their co-occurrence sets or vectors. For instance, early work by Grefenstette (1992) used the Jaccard similarity coefficient of the two sets $T(w)$ and $T(w_0)$ (cf. Equation (4) in Figure 1). Lee (1999) reviews a wide range of similarity functions, including Jaccard and the cosine between two vectors $\vec{T}(w)$ and $\vec{T}(w_0)$ (cf. Equation (5) in Figure 1).

In the context of lexical semantics, the similarity measure defined by Lin (1998) has been very successful. This measure (cf. Equation (6) in Figure 1) takes into account syntactic dependencies (d) in its co-occurrence model. In this case, the set $T(w)$ of co-occurrences of w contains pairs (d,v) of dependencies and words, representing the fact

$$sim_{Jac}(w, w_0) = \frac{|T(w) \cap T(w_0)|}{|T(w) \cup T(w_0)|} \tag{4}$$

$$sim_{cos}(w, w_0) = \frac{\sum_{i=1}^{n} \vec{T}_i(w)\vec{T}_i(w_0)}{\sqrt{\sum_{i=1}^{n} \vec{T}_i(w)^2}\sqrt{\sum_{i=1}^{n} \vec{T}_i(w_0)^2}} \tag{5}$$

$$sim_{Lin}(w, w_0) = \frac{\sum_{(d,v)\in T(w)\cap T(w_0)} (I(w, d, v) + I(w_0, d, v))}{\sum_{(d,v)\in T(w)} I(w, d, v) + \sum_{(d,v)\in T(w_0)} I(w_0, d, v)} \tag{6}$$

Figure 1
Similarity measures used in the paper. *Jac* and *cos* stand for Jaccard and cosine similarity metrics. $T(w)$ is the set of words co-occurring with w, $\vec{T}_i(w)$ is the weight of the ith element of the vector of words co-occurring with w, and $I(w, d, v)$ is the mutual information between w and d, v.

that the corpus contains an occurrence of w having dependency d with v. For instance, if the corpus contains *John loves Mary*, then the pair `(ncsubj, love)` would be in the set T for *John*. The measure uses information-theoretic principles, and $I(w, d, v)$ represents the information content of the triple (Lin 1998).

Although the use of co-occurrence vectors for words to compute similarity has been standard practice, some authors have argued for more complex uses. Schütze (1998) builds vectors for each context of occurrence of a word, combining the co-occurrence vectors for each word in the context. The vectors for contexts were used to induce senses and to improve information retrieval results. Edmonds (1997) built a lexical co-occurrence network, and applied it to a lexical choice task. Chakraborti et al. (2007) used transitivity over co-occurrence relations, with good results on several classification tasks. Note that all these works use *second order* and *higher order* to refer to their method. In this paper, we will also use *second order* to refer to a new method which goes beyond the usual co-occurrence vectors (cf. Section 3.3).

A full review of distributional models is out of the scope of this paper, as we are interested in showing that some of those models can be used successfully to improve SRC. Padó and Lapata (2007) present a review of distributional models for word similarity, and a study of several parameters that define a broad family of distributional similarity models, including Jaccard and Lin. They provide publicly available software,[4] which we have used in this paper, as explained in the next section. Baroni and Lenci (2010) present a framework for extracting distributional information from corpora that can be used to build models for different tasks.

Distributional similarity models were first used to tackle syntactic ambiguity. For instance, Pantel and Lin (2000) obtained very good results on PP-attachment using the distributional similarity measure defined by Lin (1998). Distributional similarity was used to overcome sparsity problems: Alongside the counts in the training data of the target words, the counts of words similar to the target ones were also used. Although not made explicit, Lin was actually using a distributional similarity model of selectional preferences.

The application of distributional selectional preferences to semantic roles (as opposed to syntactic functions) is more recent. Gildea and Jurafsky (2002) are the only ones applying selectional preferences in a real SRL task. They used distributional clustering and WordNet-based techniques on a SRL task on FrameNet roles. They report a very small improvement of the overall performance when using distributional clustering techniques. In this paper we present complementary experiments, with a different role set and annotated corpus (PropBank), a wider range of selectional preference models, and the analysis of out-of-domain results.

Other papers applying semantic preferences in the context of semantic roles rely on the evaluation of artificial tasks or human plausibility judgments. Erk (2007) introduced a distributional similarity–based model for selectional preferences, reminiscent of that of Pantel and Lin (2000). Her approach models the selectional preference $SP_{sim}(p, r, w_0)$ of an argument position r of governing predicate p for a possible head-word w_0 as follows:

$$SP_{sim}(p, r, w_0) = \sum_{w \in Seen(p,r)} sim(w_0, w) \cdot weight(p, r, w) \tag{7}$$

4 `http://www.coli.uni-saarland.de/~pado/dv/dv.html`.

where $sim(w_0, w)$ is the similarity between the seen and potential heads, $Seen(p, r)$ is the set of heads of role r for predicate p seen in the training data set (as in the lexical model), and $weight(p, r, w)$ is the weight of the seen head word w. Our distributional model for selectional preferences follows her formalization.

Erk instantiated the basic model with several corpus-based distributional similarity measures, including Lin's similarity, Jaccard, and cosine (Figure 1) among others, and several implementations of the weight function such as the frequency. The quality of each model instantiation, alongside Resnik's model and an expectation maximization (EM)-based clustering model, was tested in a pseudo-disambiguation task where the goal was to distinguish an attested filler of the role and a randomly chosen word. The results over 100 frame-specific roles showed that distributional similarities attain similar error rates to Resnik's model but better than EM-based clustering, with Lin's formula having the smallest error rate. Moreover, the coverage of distributional similarity measures was much better than Resnik's. In a more recent paper, Erk, Padó, and Padó (2010) extend the aforementioned work, including evaluation to human plausibility judgments and a model for inverse selectional preferences.

In this paper we test similar techniques to those presented here, but we evaluate selectional preference models in a setting directly related to semantic role classification, namely, given a selectional preference model for a verb we find the role which fits best the given head word. The problem is indeed qualitatively different from previous work in that we do not have to choose among the head words competing for a role but among selectional preferences of roles competing for a head word.

More recent work on distributional selectional preference has explored the use of discriminative models (Bergsma, Lin, and Goebel 2008) and topical models (Ó Séaghdha 2010; Ritter, Mausam, and Etzioni 2010). These models would be a nice addition to those implemented in this paper, and if effective, they would improve further our results with respect to the baselines which don't use selectional preferences.

Contrary to WordNet-based models, distributional preferences do not rely on a hand-built resource. Their coverage and generalization ability depend on the corpus from which the distributional similarity model was computed. This fact makes this approach more versatile in domain adaptation scenarios, as more specific and test-set focused generalization corpora could be used to modify, enrich, or even replace the original corpus.

2.3 PropBank

In this work we use the semantic roles defined in PropBank. The Proposition Bank (Palmer, Gildea, and Kingsbury 2005) emerged as a primary resource for research in SRL. It provides semantic role annotation for all verbs in the Penn Treebank corpus. PropBank takes a "theory-neutral" approach to the designation of core semantic roles. Each verb has a frameset listing its allowed role labelings in which the arguments are designated by number (starting from 0). Each numbered argument is provided with an English language description specific to that verb. The most frequent roles are Arg0 and Arg1 and, generally, Arg0 stands for the prototypical agent and Arg1 corresponds to the prototypical patient or theme of the proposition. The rest of arguments (Arg2 to Arg5) do not generalize across verbs, that is, they have verb specific interpretations.

Apart from the core numbered roles, there are 13 labels to designate adjuncts: AM-ADV (general-purpose), AM-CAU (cause), AM-DIR (direction), AM-DIS (discourse marker), AM-EXT (extent), AM-LOC (location), AM-MNR (manner), AM-MOD

Table 1
Example of **verb**-*role* lexical SP models for *write*, listed in alphabetical order. Number of heads indicates the number of head words attested, Unique heads indicates the number of distinct head words attested, and Examples lists some of the heads in alphabetical order.

Verb-*role*	Number of heads	Unique heads	Examples
write-*Arg0*	98	84	*Angrist anyone baker ball bank Barlow Bates ...*
write-*Arg1*	97	69	*abstract act analysis article asset bill book ...*
write-*Arg2*	7	7	*bank commander hundred jaguar Kemp member ...*
write-*AM-LOC*	2	2	*paper space*
write-*AM-TMP*	1	1	*month*

(modal verb), AM-NEG (negation marker), AM-PNC (purpose), AM-PRD (predication), AM-REC (reciprocal), and AM-TMP (temporal).

3. Selectional Preference Models for Argument Classification

Our approach for applying selectional preferences to semantic role classification is discriminative. That is, the SP-based models provide a score for every possible role label given a verb (or preposition), the head word of the argument, and the selectional preferences for the verb (or preposition). These scores can be used to directly assign the most probable role or to codify new features to train enriched semantic role classifiers.

In this section we first present all the variants for acquiring selectional preferences used in our study, and then present the method to apply them to semantic role classification. We selected several variants that have been successful in some previous works.

3.1 Lexical SP Model

In order to implement the lexical model we gathered all heads w of arguments filling a role r of a predicate p and obtained $freq(p, r, w)$ from the corresponding training data (cf. Section 4.1). Table 1 shows a sample of the heads of arguments attested in the corpus for the verb *write*. The lexical SP model can be simply formalized as follows:

$$SP_{lex}(p, r, w_0) = freq(p, r, w_0) \tag{8}$$

3.2 WordNet-Based SP Models

We instantiated the model based on (Resnik 1993b) presented in the previous section (SP_{Res}, cf. Equation (3)) using the implementation of Agirre and Martinez (2001). Tables 2 and 3 show the synsets[5] that generalize best the head words in Table 1 for **write**-*Arg0* and **write**-*Arg1*, according to the weight assigned to those synsets by Equation (1). According to this model, and following basic intuition, the words attested as being Arg0s of *write* are best generalized by semantic classes such as living things,

5 The WordNet terminology for concepts is synset. In this paper we use concept, synset, and semantic class interchangeably.

Table 2
Excerpt from the selectional preferences for **write**-*Arg0* according to SP_{Res}, showing the synsets that generalize best the head words in Table 1. Weight lists the weight assigned to those synsets by Equation (1). Description includes the words and glosses in the synset.

Synset	Weight	Description
n#00002086	5.875	life_form organism being living_thing *any living entity*
n#00001740	5.737	entity something *anything having existence* (*living or nonliving*)
n#00009457	4.782	object physical_object *a physical (tangible and visible) entity*;
n#00004123	4.351	person individual someone somebody mortal human soul *a human being*;

Table 3
Excerpt from the selectional preferences for **write**-*Arg1* according to SP_{Res}, showing the synsets that generalize best the head words in Table 1. Weight lists the weight assigned to those synsets by Equation (1). Description includes the words and glosses in the synset.

Synset	Weight	Description
n#00019671	7.956	communication *something that is communicated between people or groups*
n#04949838	4.257	message content subject_matter substance *what a communication that ...*
n#00018916	3.848	relation *an abstraction belonging to or characteristic of two entities*
n#00013018	3.574	abstraction *a concept formed by extracting common features from examples*

entities, physical objects, and human beings, whereas Arg1s by communication, message, relation, and abstraction.

Resnik's method performs well among Wordnet-based methods, but we realized that it tends to overgeneralize. For instance, in Table 2, the concept for "entity" (one of the unique beginners of the WordNet hierarchy) has a high weight. This means that a head like "grant" would be assigned Arg0. In fact, any noun which is under concept n#00001740 (*entity*) but not under n#04949838 (*message*) would be assigned Arg0. This observation led us to speculate on an alternative method which would try to generalize as little as possible.

Our intuition is that general synsets can fit several selectional preferences at the same time. For instance, the `<entity>` class, as a superclass of most words, would be a correct generalization for the selectional preferences of all *agent*, *patient*, and *instrument* roles of a predicate like *break*. On the contrary, specific concepts are usually more useful for characterizing selectional preferences, as in the `<tool>` class for the *instrument* role of *break*. The priority of using specific synsets over more general ones is, thus, justified in the sense that they may better represent the most relevant semantic characteristics of the selectional preferences.

The alternative method (SP_{wn}) is based on the *depth* of the concepts in the WordNet hierarchy and the *frequency* of the nouns. The use of the depth in hierarchies to model the specificity of concepts (the deeper the more specific) is not new (Rada et al. 1989; Sussna 1993; Agirre and Rigau 1996). Our method tries to be conservative with respect to generalization: When we check which SP is a better fit for a given target head, we always prefer the SP that contains the most specific generalization for the target head (the lowest synset which is a hypernym of the target word).

Table 4
Excerpt from the selectional preferences for **write**-*Arg0* according to SP_{wn}, showing from deeper to shallower the synsets in WordNet which are connected to head words in Table 1. Depth lists the depth of synsets in WordNet. Description includes the words and glosses in the synset.

Synset	Depth	Freq.	Description
n#01967203	9	1	humanoid human_being *any living or extinct member of the ...*
n#07603319	8	1	spy undercover_agent *a secret agent hired by a state to ...*
n#07151308	8	1	woman *a human female who does housework*
n#06183656	8	1	Federal_Reserve *the central bank of the US*

Table 5
Excerpt from the selectional preferences for **write**-*Arg1* according to SP_{wn}, showing from deeper to shallower the synsets in WordNet which are connected to head words in Table 1. Depth lists the depth of synsets in WordNet. Description includes the words and glosses in the synset.

Synset	Depth	Freq.	Description
n#05403815	13	1	information *formal accusation of a crime*
n#05401516	12	1	accusation accusal *a formal charge of wrongdoing brought ...*
n#04925620	11	1	charge complaint *a pleading describing some wrong or offense*
n#04891230	11	1	memoir *an account of the author's personal experiences*

More concretely, we model selectional preferences as a multiset[6] of synsets, storing all hypernyms of the heads seen in the training data for a certain role of a given predicate, that is:

$$S_{mul}(p,r) = \biguplus_{w \in Seen(p,r)} hyp(w) \tag{9}$$

where $Seen(p,r)$ are all the argument heads for predicate p and role r, and $hyp(w)$ returns all the synsets and hypernyms of w, including hypernyms of hypernyms recursively up to the top synsets.

For any given synset s, let $d(s)$ be the depth of the synset in the WordNet hierarchy, and let $\mathbf{1}_{S_{mul}(p,r)}(s)$ be the multiplicity function which returns how many times s is contained in the multiset $S_{mul}(p,r)$. We define a partial order among synsets $a, b \in S_{mul}(p,r)$ as follows: $ord(a) > ord(b)$ iff $d(a) > d(b)$ or $d(a) = d(b) \wedge \mathbf{1}_{S_{mul}(p,r)}(a) > \mathbf{1}_{S_{mul}(p,r)}(b)$. Tables 4 and 5 show the most specific synsets (according to their depth) for **write**-*Arg0* and **write**-*Arg1*.

We can then measure the goodness of fit of the selectional preference for a word as the rank in the partial order of the first hypernym of the head that is also present in the selectional preference. For that, we introduce $SP_{wn}(p,r,w)$, which following the previous notation is defined as:

$$SP_{wn}(p,r,w) = \arg \max_{s \in hyp(w) \cap S_{mul}(p,r)} ord(s) \tag{10}$$

6 Multisets are similar to sets, but allow for repeated members.

Table 6
Most similar words for *Texas* and *December* according to Lin (1998).

Texas	Florida 0.249, Arizona 0.236, California 0.231, Georgia 0.221, Kansas 0.217, Minnesota 0.214, Missouri 0.214, Michigan 0.213, Colorado 0.208, North Carolina 0.207, Oklahoma 0.207, Arkansas 0.205, Alabama 0.205, Nebraska 0.201, Tennessee 0.197, New Jersey 0.194, Illinois 0.189, Virginia 0.188, Kentucky 0.188, Wisconsin 0.188, Massachusetts 0.184, New York 0.183
December	June 0.341, October 0.340, November 0.333, April 0.330, February 0.329, September 0.328, July 0.323, January 0.322, August 0.317, may 0.305, March 0.250, Spring 0.147, first quarter 0.135, mid-December 0.131, month 0.130, second quarter 0.129, mid-November 0.128, fall 0.125, summer 0.125, mid-October 0.121, autumn 0.121, year 0.121, third quarter 0.119

In case of ties, the role coming first in alphabetical order would be returned. Note that, similar to the Resnik model (cf. Section 2.1), this model implicitly deals with the word ambiguity problem.

As with any other approximation to measure specificity of concepts, the use of depth has some issues, as some deeply rooted stray synsets would take priority. For instance, Table 4 shows that synset n#01967203 for *human being* is the deepest synset. In practice, when we search the synsets of a target word in the SP_{wn} models following Eq. (10), the most specific synsets (specially stray synsets) are not found, and synsets higher in the hierarchy are used.

3.3 Distributional SP Models

All our distributional SP models are based on Equation (7). We have used several variants for $sim(w_0, w)$, as presented subsequently, but in all cases, we used the frequency $freq(p, r, w)$ as the weight in the equation. Given the availability of public resources for distributional similarity, rather than implementing $sim(w_0, w)$ afresh we used (1) the precompiled similarity measures by Lin (1998),[7] and (2) the software for semantic spaces by Padó and Lapata (2007).

In the first case, Lin computed the similarity numbers for an extensive vocabulary based on his own similarity formula (cf. Equation (6) in Figure 1) run over a large parsed corpus comprising journalism texts from different sources: WSJ (24 million words), San Jose Mercury (21 million words) and AP Newswire (19 million words). The resource includes, for each word in the vocabulary, its most similar words with the similarity weight. In order to get the similarity for two words, we can check the entry in the thesaurus for either word. We will refer to this similarity measure as sim^{pre}_{Lin}. Table 6 shows the most similar words for *Texas* and *December* according to this resource.

For the second case, we applied the software to the British National Corpus to extract co-occurrences, using the optimal parameters as described in Padó and Lapata (2007, page 179): word-based space, medium context, log-likelihood association, and

7 `http://www.cs.ualberta.ca/~lindek/downloads.htm`.

Table 7
Summary of distributional similarity measures used in this work.

	Similarity measure	Source
sim_{cos}	cosine	BNC
sim_{Jac}	Jaccard	BNC
sim_{Lin}	Lin	BNC
sim^{pre}_{Lin}	Lin	Pre-computed
$sim^{pre}_{Lin\times cos}$	cosine (2nd order)	Pre-computed
$sim^{pre}_{Lin\times Jac}$	Jaccard (2nd order)	Pre-computed

2,000 basis elements. We tested Jaccard, cosine, and Lin's measure for similarity, yielding sim_{Jac}, sim_{cos}, and sim_{Lin}, respectively.

In addition to measuring the similarity of two words directly, that is, using the co-occurrence vectors of each word as in Section 2, we also tried a variant which we will call **second-order similarity**. In this case each word is represented by a vector which contains all similar words with weights, where those weights come from first order similarity. That is, in order to obtain the second-order vector for word w, we need to compute its first order similarity with all other words in the vocabulary. The second-order similarity of two words is then computed according to those vectors. For this, we just need to change the definition of T and $\vec{T}$ in the similarity formulas in Figure 1: Now $T(w)$ would return the list of words which are taken to be similar to w, and $\vec{T}(w)$ would return the same list but as a vector with weights.

This approximation is computationally expensive, as we need to compute the square matrix of similarities for all word pairs in the vocabulary, which is highly time-consuming. Fortunately, the pre-computed similarity scores of Lin (1998) (which use sim_{Lin}) are readily available, and thus the second-order similarity vectors can be easily computed. We used Jaccard and cosine to compute the similarity of the vectors, and we will refer to these similarity measures as $sim^{pre}_{Lin\times Jac}$ and $sim^{pre}_{Lin\times cos}$ hereinafter. Due to the computational complexity, we did not compute second order similarity for the semantic space software of Padó and Lapata (2007).

Table 7 summarizes all similarity measures used in this study, and the corpus or pre-computed similarity list used to build them.

3.4 Selectional Preferences for Prepositions

All the previously described models have been typically applied to **verb**-*role* selectional preferences for NP arguments. Applying them to general semantic role labeling may not be straightforward, however, and may require some extensions and adaptations. For instance, not all argument candidates are noun phrases. Common arguments with other syntactic types include prepositional, adjectival, adverbial, and verb phrases. Any candidate argument without a nominal head cannot be directly treated by the models described so far.

Table 8
Example of **prep**-*role* lexical models for the preposition *from*, listed in alphabetical order.

Prep-*role*	Number of heads	Unique heads	Examples
from-*Arg0*	32	30	*Abramson agency association barrier cut ...*
from-*Arg1*	173	118	*accident ad agency appraisal arbitrage ...*
from-*Arg2*	708	457	*academy account acquisition activity ad ...*
from-*Arg3*	396	165	*activity advertising agenda airport ...*
from-*Arg4*	5	5	*europe Golenbock system Vizcaya west*
from-*AM-ADV*	19	17	*action air air conception datum everyone ...*
from-*AM-CAU*	5	4	*air air design experience exposure*
from-*AM-DIR*	79	71	*agency alberta amendment america arson ...*
from-*AM-LOC*	20	17	*agency area asia body bureau orlando ...*
from-*AM-MNR*	29	28	*agency Carey company earnings floor ...*
from-*AM-TMP*	33	21	*april august beginning bell day dec. half ...*

A particularly interesting case is that of prepositional phrases.[8] Prepositions define relations between the preposition attachment point and the preposition complement. Prepositions are ambiguous with respect to these relations, which allows us to talk about preposition **senses**. The Preposition Project (Litkowski and Hargraves 2005, 2006) is an effort that produced a detailed sense inventory for English prepositions, which was later used in a preposition sense disambiguation task at SemEval-2007 (Litkowski and Hargraves 2007). Sense labels are defined as semantic relations, similar to those of semantic role labels. In a more recent work, Srikumar and Roth (2011) presented a joint model for extended semantic role labeling in which they show that determining the sense of the preposition is mutually related to the task of labeling the argument role of the prepositional phrase. Following the previous work, we also think that prepositions define implicit selectional preferences, and thus decided to explore the use of prepositional preferences with the aim of improving the selection of the appropriate semantic roles. Addressing other arguments with non-nominal heads has been intentionally left for further work.

The most straightforward way of including prepositional information in SP models would be to add the preposition as an extra parameter of the SP. Initial experiments revealed sparseness problems with collecting the ⟨verb, preposition, NP-head, role⟩ 4-tuples from the training set. A simpler approach consists of completely disregarding the verb information while collecting the prepositional preferences. That is, the selectional preference for a preposition p and role r is defined as the union of all nouns w found as heads of noun phrases embedded in prepositional phrases headed by p and labeled with semantic role r. Then, one can apply any of the variants described in the previous sections to calculate $SP(p, r, w)$. Table 8 shows a sample of the lexical model for the preposition *from*, organized according to the roles it plays.

These simple **prep**-*role* preferences largely avoided the sparseness problem while still being able to capture relevant information to distinguish the appropriate roles in many PP arguments. In particular, they proved to be relevant to distinguish between adjuncts of the type "[*in* New York]$_{Location}$" vs. "[*in* Winter]$_{Temporal}$." Nonetheless, we

8 Prepositional phrase is the second most frequent type of syntactic constituent for semantic arguments (13%), after noun phrases (45%).

are aware that not taking into account verb information also introduces some limitations. In particular, the simplification could damage the performance on PP core arguments, which are verb-dependent.[9] For instance, our prepositional preferences would not be able to suggest appropriate roles for the following two PP arguments: "increase [*from* seven cents a share]$_{Arg3}$" and "receive [*from* the funds]$_{Arg2}$," because the two head nouns (*cents* and *funds*) are semantically very similar. Assigning the correct roles in these cases clearly depends on the information carried by the verbs. *Arg3* is the *starting point* for the predicate *increase*, whereas *Arg2* refers to the *source* for *receive*.

Our perspective on making this simple definition of **prep**-*role* SPs was practical and just a starting point to play with the argument preferences introduced by prepositions. A more complex model, distinguishing between prepositional phrases in adjunct and core argument positions, should be able to model the linguistics better yet alleviate the sparseness problem, and would hopefully produce better results.

The combination scheme for applying **verb**-*role* and **prep**-*role* is also very simple. Depending on the syntactic type of the argument we apply one or the other model, both in learning and testing:

- When the argument is a noun phrase, we use **verb**-*role* selectional preferences.
- When the argument is a prepositional phrase, we use **prep**-*role* selectional preferences.

We thus use a straightforward method to combine both kinds of SPs. More complex possibilities like doing mixtures of both SPs are left for future work.

3.5 Role Classification with SP Models

Selectional preference models can be directly used to perform role classification. Given a target predicate p and noun phrase candidate argument with head w, we simply select the role r of the predicate which best fits the head according to the SP model. This selection rule is formalized as:

$$ROLE(p, w) = \arg \max_{r \in Roles(p)} SP(p, r, w) \tag{11}$$

with $Roles(p)$ being the set of all roles applicable to the predicate p, and $SP(p, r, w)$ the goodness of fit of the selectional preference model for the head w, which can be instantiated with all the variants mentioned in the previous subsections, including the lexical model (Equation (8)) WordNet-based SP models (Equations (3) and (10)), and distributional SP models (Equation (7)), using different similarity models as in Table 7. Ties were broken returning the role coming first according to alphabetical order. Note that in the case of SP_{wn} (Equation 10) we need to use arg min rather than arg max.

9 The percentage of prepositional phrases in core argument position is 48%, slightly lower than in adjunct position (52%).

Note that if the candidate argument is a prepositional phrase with preposition p' and embedded NP head word w, the classification rule uses the **prep**-*role* SP model, that is:

$$ROLE(p, p', w) = \arg \max_{r \in Roles(p')} SP(p', r, w)$$

4. Experiments with Selectional Preferences in Isolation

In this section we evaluate the ability of selectional preference models to discriminate among different roles. For that, SP models will be used in isolation, according to the classification rule in Equation (11), to predict role labels for a set of (*predicate*, *argument-head*) pairs. That is, we are interested in the discriminative power of the semantic information carried by the SPs, factoring out any other feature commonly used by the state-of-the-art SRL systems. The data sets used and the experimental results are presented in the following.

4.1 Data Sets

The data used in this work are the benchmark corpus provided by the CoNLL-2005 shared task on SRL (Carreras and Màrquez 2005). The data set, of over 1 million tokens, comprises PropBank Sections 02–21 for training, and Sections 24 and 23 for development and testing, respectively. The Selectional Preferences implemented in this study are not able to deal with non-nominal argument heads, such us those of NEG, DIS, MOD (i.e., SPs never predict NEG, DIS, or MOD roles); but, in order to replicate the same evaluation conditions of typical PropBank-based SRL experiments all arguments are evaluated. That is, our SP models don't return any prediction for those, and the evaluation penalizes them accordingly.

The predicate–role–head triples (p, r, w) for generalizing the selectional preferences are extracted from the arguments of the training set, yielding 71,240 triples, from which 5,587 different predicate-role selectional preferences (p, r) are derived by instantiating the different models in Section 3. Tables 9 and 10 show additional statistics about some of the most (and least) frequent verbs and prepositions in these tuples.

The test set contains 4,134 pairs (covering 505 different predicates) to be classified into the appropriate role label. In order to study the behavior on out-of-domain data, we also tested on the PropBanked part of the Brown corpus (Marcus et al. 1994). This corpus contains 2,932 (p, w) pairs covering 491 different predicates.

4.2 Results

The performance of each selectional preference model is evaluated by calculating the customary precision (P), recall (R), and F_1 measures.[10] For all experiments reported in this paper, we checked for statistical significance using bootstrap resampling (100 samples) coupled with one-tailed paired t-test (Noreen 1989). We consider a result significantly better than another if it passes this test at the 99% confidence interval.

10 $P = Correct/Predicted * 100$, $R = Correct/Gold * 100$, where *Correct* is the number of correct predictions, *Predicted* is the number of predictions, and *Gold* is the total number of gold annotations.
$F_1 = 2PR/(P + R)$ is the harmonic mean of P and R.

Table 9
Statistics of the three most and least frequent verbs in the training set. Role frame lists the types of arguments seen in training for each verb; Heads indicates the total number of arguments for the verb; Heads per role shows the average number of head words for each role; and Unique heads per role lists the average number of unique head words for each verb's role.

Verb	Role frame	Heads	Heads per role	Unique heads per role
say	Arg0,Arg1,Arg3,AM-ADV, AM-LOC, AM-MNR, AM-TMP, AM-LOC,AM-MNR	7,488	1,069	371
have	Arg0,Arg1,AM-ADV,AM-LOC AM-MNR,AM-NEG,AM-TMP	3,487	498	189
make	Arg0,Arg1,Arg2,AM-ADV AM-LOC,AM-MNR,AM-TMP	2,207	315	143
...	...	...	...	...
accrete	Arg1	1	1	1
accede	Arg0	1	1	1
absolve	Arg0	1	1	1

Table 10
Statistics of the three most and least frequent prepositions in the training set. Role frame lists the types of arguments seen in training for each preposition; Heads indicates the total number of arguments for the preposition; Heads per role shows the average number of head words for each role; and Unique heads per role lists the average number of unique head words for each preposition's role.

Preposition	Role frame	Heads	Heads per role	Unique heads per role
in	Arg0,Arg1,Arg2,Arg3,Arg4,Arg5 AM-ADV,AM-CAU,AM-DIR,AM-DIS, AM-EXT,AM-LOC,AM-MNR,AM-NEG, AM-PNC,AM-PRD,AM-TMP	6,859	403	81
to	Arg0,Arg1,Arg2,Arg3,Arg4, AM-ADV,AM-CAU,AM-DIR,AM-DIS, AM-EXT,AM-LOC,AM-MNR,AM-PNC, AM-PRD,AM-TMP	3,495	233	94
for	Arg0,Arg1,Arg2,Arg3,Arg4, AM-ADV,AM-CAU,AM-DIR,AM-DIS, AM-LOC,AM-MNR,AM-PNC,AM-TMP	2,935	225	74
...	...	...	...	...
beside	Arg2, AM-LOC	2	1	1
atop	Arg2, AM-DIR	2	1	1
aboard	AM-LOC	1	1	1

Tables 11 and 12 list the results of the various selectional preference models in isolation. Table 11 shows the results for **verb**-*role* SPs, and Table 12 lists the results for the combination of **verb**-*role* and **preposition**-*role* SPs as described in Section 3.4.[11] It is worth noting that the results of Tables 11 and 12 are calculated over exactly the

11 Note that the results reported here are not identical to those we reported in Zapirain, Agirre, and Màrquez (2009). The differences are two-fold: (a) in our previous experiments we discarded roles such as MOD, DIS, and NEG, whereas here we evaluate on all roles, and (b) our previous work used only the subset of the data that could be mapped to VerbNet (around 50%), whereas here we inspect all tuples.

Table 11
Results for **verb**-*role* SPs in the development partition of WSJ, the test partition of WSJ, and the Brown corpus. For each experiment, we show precision (P), recall (R), and F_1. Values in boldface font are the highest in the corresponding column. F_1 values marked with † are significantly lower than the highest F_1 score in the same column.

	Verb-*role* SPs								
	Development			WSJ Test			Brown Test		
	P	R	F_1	P	R	F_1	P	R	F_1
lexical	**73.94**	21.81	33.69†	**70.75**	26.66	39.43†	**59.39**	05.51	10.08†
SP_{Res}	43.65	35.70	39.28†	45.07	37.11	40.71†	36.34	27.58	31.33†
SP_{wn}	53.09	43.35	47.73†	55.44	45.58	50.03†	41.76	31.58	35.96†
$SP_{sim_{Lin}}$	53.88	44.35	48.65†	52.27	45.13	48.66†	48.30	32.08	38.56†
$SP_{sim_{Jac}}$	48.40	45.53	46.92†	48.85	46.38	47.58†	42.10	34.34	37.82†
$SP_{sim_{cos}}$	52.37	49.26	50.77†	53.13	50.44	51.75†	43.24	35.27	38.85†
$SP_{sim^{pre}_{Lin}}$	60.29	**59.54**	**59.91**	59.93	**59.38**	59.65	50.79	**48.39**	**49.56**
$SP_{sim^{pre}_{Lin \times Jac}}$	60.56	56.97	58.71	61.76	58.63	**60.16**	51.97	42.39	46.69†
$SP_{sim^{pre}_{Lin \times cos}}$	60.22	56.64	58.37	61.12	58.12	59.63	51.92	42.35	46.65†

Table 12
Results for combined **verb**-*role* and **prep**-*role* SPs in the development partition of WSJ, the test partition of WSJ, and the Brown corpus. For each experiment, we show precision (P), recall (R), and F_1. Values in boldface font are the highest in the corresponding column. F_1 values marked with † are significantly lower from the highest F_1 score in the same column.

	Preposition-*role* and **Verb**-*role* SPs								
	Development			WSJ Test			Brown Test		
	P	R	F_1	P	R	F_1	P	R	F_1
lexical	**82.05**	39.17	53.02†	**82.98**	43.77	57.31†	**68.47**	13.60	22.69†
SP_{Res}	63.72	53.09	57.93†	63.47	53.24	57.91†	55.12	44.15	49.03†
SP_{wn}	71.72	59.68	65.15†	65.70	63.88	64.78†	60.08	48.10	53.43†
$SP_{sim_{Lin}}$	63.84	54.58	58.85†	63.75	56.40	59.85†	54.27	39.96	46.04†
$SP_{sim_{Jac}}$	61.75	61.13	61.44†	61.83	61.40	61.61†	55.42	53.45	54.42†
$SP_{sim_{cos}}$	64.81	64.17	64.49†	64.67	64.22	64.44†	56.56	54.54	55.53†
$SP_{sim^{pre}_{Lin}}$	67.78	67.10	67.44†	68.34	67.87	68.10†	58.43	56.35	57.37†
$SP_{sim^{pre}_{Lin \times Jac}}$	69.90	**69.20**	**69.55**	70.82	**70.33**	**70.57**	62.37	**60.15**	**61.24**
$SP_{sim^{pre}_{Lin \times cos}}$	69.47	68.78	69.12	70.28	69.80	70.04	62.36	60.14	61.23

same example set. PP arguments are treated by the **verb**-*role* SPs by just ignoring the preposition and considering the head noun of the NP immediately embedded in the PP.

It is worth mentioning that none of the SP models is able to predict the role when facing a head word missing from the model. This is especially noticeable in the lexical model, which can only return predictions for words seen in the training data and is

penalized in recall. WordNet based models, which have a lower word coverage compared to distributional similarity–based models, are also penalized in recall.

In both tables, the lexical row corresponds to the baseline lexical match method. The following rows correspond to the WordNet-based selectional preference models. The distributional models follow, including the results obtained by the three similarity formulas on the co-occurrences extracted from the BNC (sim_{Jac}, sim_{cos} sim_{Lin}), and the results obtained when using Lin's pre-computed similarities directly (sim^{pre}_{Lin}) and as a second-order vector ($sim^{pre}_{Lin\times Jac}$ and $sim^{pre}_{Lin\times cos}$).

First and foremost, this experiment proves that splitting SPs into **verb**- and **preposition**-*role* SPs yields better results. The comparison of Tables 11 and 12 shows that the improvements are seen for both precision and recall, but especially remarkable for recall. The overall F_1 improvement is of up to 10 points. Unless stated otherwise, the rest of the analysis will focus on Table 12.

As expected, the lexical baseline attains a very high precision in all data sets, which underscores the importance of the lexical head word features in argument classification. Its recall is quite low, however, especially in Brown, confirming and extending Pradhan, Ward, and Martin (2008), who also report a similar performance drop for argument classification on out-of-domain data. All our selectional preference models improve over the lexical matching baseline in recall, with up to 24 absolute percentage points in the WSJ test data set and 47 absolute percentage points in the Brown corpus. This comes at the cost of reduced precision, but the overall F-score shows that all selectional preference models are well above the baseline, with up to 13 absolute percentage points on the WSJ data sets and 39 absolute percentage points on the Brown data set. The results, thus, show that selectional preferences are indeed alleviating the lexical sparseness problem.[12]

As an example, consider the following head words of potential arguments of the verb *wear* found in the test set: *doctor*, *men*, *tie*, *shoe*. None of these nouns occurred as heads of arguments of *wear* in the training data, and thus the lexical feature would be unable to predict any role for them. Using selectional preferences, we successfully assigned the `A0` role to *doctor* and *men*, and the `A1` role to *tie* and *shoe*.

Regarding the selectional preference variants, WordNet-based and first-order distributional similarity models attain similar levels of precision, but the former have lower recall and F_1. The performance loss on recall can be explained by the limited lexical coverage of WordNet when compared with automatically generated thesauri. Examples of words missing in WordNet include abbreviations (e.g., *Inc.*, *Corp.*) and brand names (e.g., *Texaco*, *Sony*).

The comparison of the WordNet-based models indicates that our proposal for a lighter method of WordNet-based selectional preference was successful, as our simpler variant performs better than Resnik's method. In manual analysis, we realized that Resnik's model tends to always predict the most frequent roles whereas our model covers a wider role selection. Resnik's tendency to overgeneralize makes more frequent roles cover all the vocabulary, and the weighting system penalizes roles with fewer occurrences.

12 We verified that the lexical model shows higher classification accuracy than all the more elaborate SP models on the subset of cases covered by both the lexical and the SP models. In this situation, if we aimed at constructing the best role classifier with SPs alone we could devise a back-off strategy, in the style of Chambers and Jurafsky (2010), which uses the predictions of the lexical model when present and one of the SP models if not. As presented in Section 5, however, our main goal is to integrate these SP models in a real end-to-end SRL system, so we keep their analysis as independent predictors for the moment.

The results for distributional models indicate that the SPs using Lin's ready-made thesaurus (sim^{pre}_{Lin}) outperforms Padó and Lapata's distributional similarity model (Padó and Lapata 2007) calculated over the BNC (sim_{Lin}) in both Tables 11 and 12. This might be due to the larger size of the corpus used by Lin, but also by the fact that Lin used a newspaper corpus, compared with the balanced BNC corpus. Further work would be needed to be more conclusive, and, if successful, could improve further the results of some SP models.

Among the three similarity metrics using Padó and Lapata's software, the cosine seems to perform consistently better. Regarding the comparison between first-order and second-order using pre-computed similarity models, the results indicate that second-order is best when using both the **verb**-*role* and **prep**-*role* models (cf. Table 12), although the results for **verb**-*roles* are mixed (cf. Table 11). Jaccard seems to provide slightly better results than cosine for second-order vectors.

In summary, the use of separate **verb**-*role* and **prep**-*role* models produces the best results, and second-order similarity is highly competitive. As far as we know, this is the first time that **prep**-*role* models and second-order models are applied to selectional preference modeling.

5. Semantic Role Classification Experiments

In this section we advance the use of SP in SRL one step further and show that selectional preferences are able to effectively improve performance of a state-of-the-art SRL system. More concretely, we integrate the information of selectional preference models in a SRL system and show significant improvements in role classification, especially when applied to out-of-domain corpora.[13]

We will use some of the selectional preference models presented in the previous section. We will focus on the combination of **verb**-*role* and **prep**-*role* models. Regarding the similarity models, we will choose the best two performing models from each of the three families that we tried, namely, the two WordNet models, the two best models based on the BNC corpus (sim_{Jac},sim_{cos}), and the two best models based on Lin's precomputed similarity metrics (sim^2_{Jac},sim^2_{cos}). We left the exploration of other combinations for future work.

5.1 Integrating Selectional Preferences in Role Classification

For these experiments, we modified the *SwiRL* SRL system, a state-of-the-art semantic role labeling system (Surdeanu et al. 2007). *SwiRL* ranked second among the systems that did not implement model combination at the CoNLL-2005 shared task and fifth overall (Carreras and Màrquez 2005). Because the focus of this section is on role classification, we modified the SRC component of *SwiRL* to use gold argument boundaries, that is, we assume that semantic role identification works perfectly. Nevertheless, for a realistic evaluation, all the features in the role classification model are generated using actual syntactic trees generated by the Charniak parser (Charniak 2000).

The key idea behind our approach is model combination: We generate a battery of base models using all resources available and we combine their outputs using multiple strategies. Our pool of base models contains 13 different models: The first is the

13 The data sets used for the experiments reported in this section are exactly the ones described in Section 4.1.

unmodified *SwiRL* SRC, the next six are the selected SP models from the previous section, and the last six are variants of *SwiRL* SRC. In each variant, the feature set of the unmodified *SwiRL* SRC model is extended with a single feature that models the choice of a given SP, for example, SRC+SP$_{res}$ contains an extra feature that indicates the choice of Resnik's SP model.[14]

We combine the outputs of these base models using two different strategies: (a) majority voting, which selects the label predicted by most models, and (b) meta-classification, which uses a supervised model to learn the strengths of each base model. For the meta-classification model, we opted for a binary classification approach: First, for each constituent we generate n data points, 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 either correct or incorrect. We trained the meta-classifier on the usual PropBank training partition, using 10-fold 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.

The binary meta-classifier uses the following set of features:

- *Labels proposed by the base models*, for example, the feature `SRC+SP`$_{res}$`=Arg0` indicates that the SRC+SP$_{res}$ base model proposed the Arg0 label. We add 13 such features, one for each base model. Intuitively, this feature allows the meta-classifier to learn the strengths of each base model with respect to role labels: SRC+SP$_{res}$ should be trusted for the Arg0 role, and so on.
- *Boolean value indicating agreement with the majority vote*, for example, the feature `Majority=true` indicates that the majority of the base models proposed the same label as the one currently considered by the meta-classifier.
- *Number of base models that proposed this data point's label*. To reduce sparsity, for each number of base models, N, we generate N distinct features indicating that the number of base models that proposed this label is larger than k, where $k \in [0, N)$. For example, if two base models proposed the label under consideration, we generate the following two features: `BaseModelNumber>0` and `BaseModelNumber>1`. This feature provides finer control over the number of votes received by a label than the majority voter, for example, the meta-classifier can learn to trust a label if more than two base models proposed it, even if the majority vote disagrees.
- *List of actual base models that proposed this data point's label*. We store a distinct feature for each base model that proposed the current label, and also a concatenation of all these base model names. The latter feature is designed to allow the meta-classifier to learn preferences for certain combinations of base models. For example, if two base models, SP_{res} and SP_{wn}, proposed the label under consideration, we generate three features: `Base=`SP_{res}, `Base=`SP_{wn}, and `Base=`SP_{res}+SP_{wn}.

14 Adding more than one SP output as a feature in *SwiRL*'s SRC model did not improve performance in development over the single-SP SRC model. Our conjecture is that the large number of features in SRC has the potential to *drown* the SP-based features. This may be accentuated when there are more SP-based features because their signal is divided among them due to their overlap. We have also tried to add the input features of the SP models directly to the SRC model but this also proved to be unsuccessful during development.

Table 13
Results for the combination approaches. Accuracy shows the overall results. Core and Adj contain F_1 results restricted to the core numbered roles and adjuncts, respectively. SRC is *SwiRL*'s standalone SRC model; +SP$_x$ stands for the SRC model extended with a feature given by the corresponding SP model. Values in boldface font are the highest in the corresponding column. Accuracy values marked with † are significantly lower than the highest accuracy score in the same column.

	WSJ test			Brown test		
	Acc.	Core F_1	Adj. F_1	Acc.	Core F_1	Adj. F_1
SRC	90.83†	93.25	81.31	79.52	84.42	57.76
+SP$_{Res}$	90.76†	93.17	81.08	79.86†	84.52	59.24
+SP$_{wn}$	90.56†	92.88	81.11	79.73†	84.26	59.69
+SP$_{sim_{Jac}}$	90.86†	93.37	80.30	79.83†	84.43	59.54
+SP$_{sim_{cos}}$	90.87†	93.33	80.92	80.50†	85.14	60.16
+SP$_{sim^{pre}_{Lin \times Jac}}$	90.95†	93.03	82.75	80.75†	85.62	59.63
+SP$_{sim^{pre}_{Lin \times cos}}$	91.23†	93.78	80.56	80.48†	84.95	61.01
Meta-classifier	**92.43**	**94.62**	**84.00**	81.94	86.25	63.36
Voting	92.36	94.57	83.68	**82.15**	**86.37**	**63.78**

5.2 Results for Semantic Role Classification

Table 13 compares the performance of both combination approaches against the standalone SRC model. In the table, the SRC+SP$_*$ models stand for SRC classifiers enhanced with one feature from the corresponding SP. The meta-classifier shown in the table combines the output of all the 13 base models introduced previously. We implemented the meta-classifier using Support Vector Machines (SVMs)[15] with a quadratic polynomial kernel, and $C = 0.01$ (tuned in the development set).[16] Lastly, Table 13 shows the results of the voting strategy, over the same set of base models.

In the columns we show overall classification accuracy and F_1 results for both core arguments (Core) and adjunct arguments (Adj.). Note that for the overall SRC scores, we report classification accuracy, defined as ratio of correct predictions over total number of arguments to be classified. The reason for this is that the models in this section always return a label for all arguments to be classified, and thus accuracy, precision, recall, and F_1 are all equal.

Table 13 indicates that four out of the six SRC+SP$_*$ models perform better than the standalone SRC model in domain (WSJ), and all of them outperform SRC out of domain (Brown). The improvements are small, however, and, generally, not statistically significant. On the other hand, the meta-classifier outperforms the original SRC model both in domain (17.4% relative error reduction; 1.60 points of accuracy improvement) and out of domain (13.4% relative error reduction; 2.42 points of accuracy improvement), and the differences are statistically significant. This experiment proves our claim that SPs can be successfully used to improve semantic role classification. It also underscores the fact that combining SRC and SPs is not trivial, however. Our hypothesis is that this

15 `http://svmlight.joachims.org`.
16 We have also trained the meta-classifier with other learning algorithms (e.g., logistic regression with L2 regularization) and we obtained similar but slightly lower results.

is caused by the large performance disparity (20 F_1 points in domain and 18 out of domain) between the original SRC model and the standalone SP methods.

Interestingly, the meta-classifier performs only marginally better than the voting approach in domain and slightly worse out of domain. We believe that this is another effect of the above observation: Given the weaker SP-based features, the meta-classifier does not learn much beyond a majority vote, which is exactly what the simpler, unsupervised voting method models. Nevertheless, regardless of the combination method, this experiment emphasizes that infusing SP information in the SRC task is beneficial.

Table 13 also 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 (6.02 absolute points) vs. core arguments (1.83 points, for the voting model). 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.

Regarding the performance of individual combinations of SRC and SP methods (e.g., SRC+SP_{Res}), the differences among SP models in Table 13 are much smaller than in Table 12. $SP_{sim^{pre}_{Lin \times cos}}$ and $SP_{sim^{pre}_{Lin \times Jac}}$ yield the best results in both cases, and distributional methods are slightly stronger than WordNet-based methods. SP_{Res} and SP_{wn} perform similarly when combined, with a small lead for Resnik's method. The smaller differences and changes in the rank among SP methods are due to the complex interactions when combining SP models with the SRC system.

Table 14
Precision (P), recall (R), and F_1 results per argument type for the standalone SRC model and the meta-classifier, in the two test data sets (WSJ and Brown). Due to space limitations, the AM- prefix has been dropped from the labels of all adjuncts. When classifying all arguments (last row), the F_1 score is an accuracy score because in this scenario P = R = F_1. We checked for statistical significance for the overall F_1 scores (All row). Values in boldface font indicate the highest F_1 score in the corresponding row and block. F_1 values marked with † are significantly lower than the corresponding highest F_1 score.

	WSJ test						Brown test					
	SRC			Meta-classifier			SRC			Meta-classifier		
	P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1
Arg0	93.6	96.7	95.1	95.1	97.4	96.2	87.6	89.3	88.4	89.4	91.0	90.2
Arg1	93.3	94.5	93.9	94.2	95.7	95.0	84.3	90.6	87.3	86.2	91.9	89.0
Arg2	86.0	82.6	84.3	87.8	87.4	87.6	52.7	56.8	54.7	55.9	59.9	57.8
Arg3	77.6	63.4	69.8	82.4	68.3	74.7	36.4	19.0	25.0	45.8	26.2	33.3
Arg4	86.8	78.6	82.5	89.5	81.0	85.0	59.4	34.5	43.7	67.9	34.5	45.8
Core	92.9	93.6	93.3	94.2	95.1	94.6	82.6	86.3	84.4	84.6	87.9	86.3
ADV	58.5	51.4	54.7	64.4	52.3	57.7	45.1	24.3	31.6	51.9	25.7	34.4
CAU	61.1	71.0	65.7	80.0	77.4	78.7	64.7	45.8	53.7	84.6	45.8	59.5
DIR	46.2	25.0	32.4	68.8	45.8	55.0	64.7	45.8	53.7	73.9	44.5	55.6
DIS	84.3	82.7	83.5	95.6	82.7	88.7	52.6	27.0	35.7	54.5	32.4	40.7
EXT	50.0	12.5	20.0	50.0	12.5	20.0	0.0	0.0	0.0	0.0	0.0	0.0
LOC	85.2	80.9	83.0	85.0	84.7	84.8	67.8	61.2	64.3	68.3	68.7	68.5
MNR	55.8	54.1	55.0	68.9	61.7	65.1	47.4	38.9	42.7	59.2	49.3	53.8
PNC	51.9	37.8	43.8	62.5	40.5	49.2	51.7	39.5	44.8	53.3	42.1	47.1
TMP	93.6	95.9	94.7	92.8	95.9	94.4	79.0	78.1	78.5	84.1	83.2	83.7
Adj	83.1	79.6	81.3	86.2	81.9	84.0	64.9	52.1	57.8	69.8	58.0	63.4
All	–	–	90.8†	–	–	**92.4**	–	–	79.5†	–	–	**81.9**

Lastly, Table 14 shows a breakdown of the results by argument type for the original SRC model and the meta-classifier (results are also presented over all numbered arguments, Core, adjuncts, and Adj). This comparison emphasizes the previous observation that SPs are more useful for arguments that are independent of syntax than for arguments that are usually tied to certain syntactic constructs (i.e., Arg0 and Arg1). For example, in domain the meta-classifier improves Arg0 classification with 1.1 F_1 points, but it boosts the classification performance for causative arguments (AM-CAU) with 13 absolute points. A similar behavior is observed out of domain. For example, whereas Arg0 classification is improved with 1.7 points, the classification of manner arguments (AM-MNR) is improved by 11 points. All in all, with two exceptions, selectional preferences improve classification accuracy for all argument types, both in and out of domain.

The previous experiments showed that a meta-classifier (and a voting approach) over a battery of base models improves over the performance of each individual classifier. Given that half of our base models are all relatively minor changes of the same original classifier (*SwiRL*), however, it would be desirable to ensure that the overall performance gain of the meta-classification system is due to the infusion of semantic information that is missing in the baseline SRC, and not to a regularization effect coming from the ensemble of classifiers. The qualitative analysis presented in Section 6 will reinforce this hypothesis.

5.3 Results for End-to-End Semantic Role Labeling

Lastly, we investigate the contribution of SPs in an end-to-end SRL system. As discussed before, our approach focuses on argument classification, a subtask of complete SRL, because this component suffers in the presence of lexical data sparseness (Pradhan, Ward, and Martin 2008). To understand the impact of SPs on the complete SRL task we compared two *SwiRL* models: one that uses the original classification model (the SRC line in Table 13) and another that uses our meta-classifier model (the Meta-classifier line in Table 13). To implement this experiment we had to modify the publicly downloadable *SwiRL* model, which performs identification and classification jointly, using a single multi-class model. We changed this framework to a pipeline model, which first performs argument identification (i.e., is this constituent an argument or not?), followed by argument classification (i.e., knowing that this constituent is an argument, what is its label?).[17] We used the same set of features as the original *SwiRL* system and the original model to identify argument boundaries. This pipeline model allowed us to easily plug in different classification models, which offers a simple platform to evaluate the contribution of SPs in an end-to-end SRL system.

Table 15 compares the original *SwiRL* pipeline (*SwiRL* in the table) with the pipeline model where the classification component was replaced with the meta-classifier previously introduced (*SwiRL* w/ meta). The latter model backs off to the original classification model for candidates that are not covered by our current selectional preferences (i.e., are not noun phrases or prepositional phrases containing a noun phrase as the second child). We report results for the test partitions of WSJ and Brown in the same table. Note that these results are not directly comparable with the results in Tables 13 and 14, because in those initial experiments we used gold argument boundaries whereas

17 This pipeline model performs slightly worse than the original *SwiRL* on the WSJ data and slightly better on Brown.

Table 15
Precision (P), recall (R), and F_1 results per argument for the end-to-end semantic role labeling task. We compared two models: the original *SwiRL* model and the one where the classification component was replaced with the meta-classifier introduced at the beginning of the section. We used the official CoNLL-2005 shared-task scorer to produce these results. We checked for statistical significance for the overall F_1 scores (All row). Values in boldface font indicate the highest F_1 score in the corresponding row and block. F_1 values marked with † are significantly lower than the corresponding highest F_1 score.

	WSJ test						Brown test					
	SwiRL			*SwiRL* w/ meta			*SwiRL*			*SwiRL* w/ meta		
	P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1
Arg0	87.0	81.6	84.2	87.8	81.9	84.8	86.6	81.3	83.9	87.3	81.7	84.4
Arg1	79.1	71.8	75.3	79.4	72.1	75.6	70.2	64.6	67.3	71.1	65.2	68.0
Arg2	70.0	56.6	62.6	69.2	58.3	63.3	41.8	42.7	42.2	42.3	44.6	43.4
Arg3	72.4	43.9	54.7	72.6	44.5	55.2	36.4	12.9	19.0	34.6	14.5	20.5
Arg4	73.3	61.8	67.0	73.8	60.8	66.7	48.8	25.6	33.6	44.4	25.6	32.5
ADV	59.4	50.6	54.6	59.5	50.0	54.4	49.0	38.2	42.9	49.9	38.5	43.5
CAU	61.5	43.8	51.2	66.0	45.2	53.7	58.7	35.5	44.3	59.1	34.2	43.3
DIR	44.7	20.0	27.6	50.0	22.6	30.9	59.0	27.2	37.2	61.3	25.9	36.5
DIS	76.1	63.8	69.4	77.0	63.8	69.7	58.8	41.0	48.3	59.7	41.3	48.9
EXT	72.7	50.0	59.3	72.7	50.0	59.3	20.0	8.1	11.5	21.4	8.1	11.8
LOC	64.7	52.9	58.2	64.8	55.4	59.7	48.3	37.7	42.3	46.8	40.5	43.5
MNR	59.1	52.0	55.3	61.4	51.7	56.2	53.8	47.3	50.3	55.9	48.3	51.8
PNC	47.1	34.8	40.0	46.4	33.9	39.2	51.8	26.4	35.0	52.4	26.7	35.1
TMP	78.7	71.4	74.9	78.4	71.5	73.8	59.7	60.6	60.2	61.0	61.2	61.1
All	79.7	70.9	75.0†	80.0	71.3	**75.4**	71.8	64.2	67.8†	72.4	64.6	**68.4**

Table 15 shows results for an end-to-end model, which includes predicted argument boundaries.

Table 15 shows that the use of selectional preferences improves overall results when using predicted argument boundaries as well. Selectional preferences improve F_1 scores for four out of five core arguments in both WSJ and Brown, for six out of nine modifier arguments in WSJ, and for seven out of nine modifier arguments in Brown. Notably, the SPs improve results for the most common argument types (Arg0 and Arg1). All in all, SPs yield a 0.4 F_1 point improvement in WSJ and 0.6 F_1 point improvement in Brown. These improvements are small but they are statistically significant. We consider these results encouraging, especially considering that only a small percentage of arguments are actually inspected by selectional preferences. This analysis is summarized in Table 16, which lists how many argument candidates are inspected by the system in its different stages. The table indicates that the vast majority of argument candidates are filtered out by the argument identification component, which does *not* use SPs. Because of this, even though approximately 50% of the role classification decisions can be reinforced with SPs, only 4.5% and 3.6% of the *total* number of argument candidates in WSJ and Brown, respectively, are actually inspected by the classification model that uses SPs.

6. Analysis and Discussion

We conducted a complementary manual analysis to further verify the usefulness of the semantic information provided by the selectional preferences. We manually inspected 100 randomly selected classification cases, 50 examples in which the meta-classifier is

Table 16
Counts for argument candidates for the two test partitions on the end-to-end semantic role labeling task. The Predicted non-arguments line indicates how many candidate arguments are classified as non-arguments by the argument identification classifier. The Incompatible with SPs line indicates how many candidates were classified as arguments but cannot be modeled by our current SPs (i.e., they are not noun phrases or prepositional phrases containing a noun phrase as the second child). Lastly, the Compatible with SPs line lists how many candidates were both classified as likely arguments and can be modeled by the SPs.

	WSJ test	Brown test
Predicted non-arguments	158,310	184,958
Incompatible with SPs	5,739	11,167
Compatible with SPs	7,691	7,867
Total	171,740	203,992

correct and the baseline SRC (*SwiRL*) is wrong, and 50 where the meta-classifier chooses the incorrect classifier and the SRC is right. Interestingly, we observed that the majority of cases have a clear linguistic interpretation, shedding light on the reasons why the meta-classifier using SP information manages to correct some erroneous predictions of the original SRC model, but also on the limitations of selectional preferences.

Regarding the success of the meta-classifier, the studied cases generally correspond to low frequency verb–argument head pairs, in which the baseline SRC might have had problems with generalization. In 29 of the cases (∼58%), the syntactic information is not enough to disambiguate the proper role, tends to indicate a wrong role label, or it confuses the SRC because it contains errors. Most of the semantically based SP predictions are correct, however, so the meta-classifier does select the correct role label. In another 15 cases (∼30%) the source of the baseline SRC error is not clear, but still, several SP models suggest the correct role, giving the opportunity to the meta-classifier to make the right choice. Finally, in the remaining six cases (∼12%) a "chance effect" is observed: The failure of the baseline SRC model does not have a clear interpretation and, moreover, most SP predictions are actually wrong. In these situations, several labels are predicted with the same confidence, and the meta-classifier selects the correct one by chance.

Figure 2 shows four real examples in which we see the importance of the information provided by the selectional preferences. In example (a), the verb *flash* never occurs in training with the argument head word **news**. The syntactic structure alone strongly suggests Arg0, because the argument is an NP just to the left of a verb in active form. This is probably why the baseline SRC incorrectly predicts Arg0. Some semantic information is needed to know that the word **news** is not the agent of the predicate (Arg0), but rather the theme (*thing shining*, Arg1). Selectional preferences make this work perfectly, because all variants predict the correct label by signaling that **news** is much more compatible with *flash* in Arg1 position rather than Arg0.

In example (b), the predicate *promise* expects a person as Arg1 (*person promised to*, Recipient) and an action as Arg2 (*promised action*, Theme). Moreover, the presence of Arg2 is obligatory. The syntactic structure is correct but does not provide the semantic (*Arg1 should be a person*) or structural information (*the assignment of Arg1 would have required an additional Arg2*) needed to select the appropriate role. *SwiRL* does not have it either, and it assigns the incorrect Arg1 label. Most SP models correctly predict that **investigation** is more similar to the heads of Arg2 arguments of *promise* than to the heads of Arg1 arguments, however.

(a) Several traders could be seen shaking their heads when (([the **news**]$_{\text{Arg0}\Rightarrow\text{Arg1}}$)$^{\text{NP}}$ (*flashed*)$^{\text{VP}}$)$^{\text{S}}$.

(b) Italian President Francesco Cossiga (*promised* ([a quick **investigation** into whether Olivetti broke Cocom rules]$_{\text{Arg1}\Rightarrow\text{Arg2}}$)$^{\text{NP}}$)$^{\text{VP}}$.

(c) Annual payments (will more than *double* ([**from** (a year ago)$^{\text{NP}}$]$_{\text{TMP}\Rightarrow\text{Arg3}}$)$^{\text{PP}}$ to about \$240 million $\cdots$)$^{\text{VP}}$ $\cdots$

(d) Procter & Gamble Co. plans to (begin ((*testing* (next month)$^{\text{NP}}$)$^{\text{VP}}$)$^{\text{S}}$ ([a superco. **detergent** that $\cdots$ washload]$_{\text{Arg0}\Rightarrow\text{Arg1}}$)$^{\text{NP}}$)$^{\text{VP}}$.

Figure 2
Examples of incorrect *SwiRL* role assignments fixed by the meta-classifier. In each sentence, the verb is emphasized in italics and the head word for the selectional preferences is boldfaced. The argument under focus is marked within square brackets. $x \Rightarrow y$ means that the incorrect label x assigned by the baseline *SwiRL* model is corrected into role label y by the combined system. Finally, examples also contain simplified syntactic annotations from the test set predicted syntactic layer, which are used for the discussion in the text.

In example (c) we see the application of **prep**-*role* selectional preferences. In that sentence, the baseline SRC is likely confused by the content word feature of the PP "**from** a year ago" (Surdeanu et al. 2003). In PropBank, "year" is a strong indicator of a temporal adjunct (AM-TMP). The predicate *double*, however, describes the Arg3 argument as "starting point" of the action and it is usually introduced by the preposition *from*. This is very common also for other motion verbs (*go*, *rise*, etc.), resulting in the **from**-*Arg3* selectional preference containing a number of heads of temporal expressions, in particular many more instances of the word *year* than the **from**-*AM-TMP* selectional preference. As a consequence, the majority of SP models predict the correct Arg3 label.

Finally, example (d) highlights that selectional preferences increase robustness in front of parsing errors. In this example, the NP "a superco. **detergent**" is incorrectly attached to "begin" instead of the predicate *testing* by the syntactic parser. This produces many incorrect features derived from syntax (syntactic frame, path, etc.) that may confuse the baseline SRC model, which ends up producing an incorrect Arg0 assignment. Most of the SP models, however, predict that **detergent** is not a plausible Agent for *test* ("examiner"), but instead it fits best with the Arg1 position ("examined").

Nevertheless, selectional preferences have a significant limitation: They do not model syntactic structures, which often give strong hints for classification. In fact, the vast majority of the situations where the meta-classifier performs worse than the original SRC model are cases that are syntax-driven, hence situations that are incompletely addressed by the current SP models. Even though the SRC and the SRC+SP models have features that model syntax, they can be overwhelmed by the SP features and standalone models, which leads to incorrect meta-classification results. Figure 3 shows a few representative examples in this category. In the first example in the figure, the meta-classifier changes the correctly assigned label Arg2 to Arg1, because most SP models favor the Arg1 label for the argument "test." In the PropBank training corpus, however, the argument following the verb *fail* is labeled Arg2 in 79% of the cases. Because the SP models do not take into account syntax or positional information, this syntactic preference is lost. Similarly, SPs do not model the fact that the verb *buy* is seldom preceded by an Arg1 argument, or the argument immediately following the verb *precede* tends to be Arg1, hence the incorrect classifications in Figure 3 (b) and (c). All these

(a) Some "circuit breakers" installed after the October 1987 crash (*failed* ([their first **test**]$_{\text{Arg2}\Rightarrow\text{Arg1}}$)$^{\text{NP}}$)$^{\text{VP}}$...

(b) Many fund managers argue that now's ([the **time**]$_{\text{TMP}\Rightarrow\text{Arg1}}$)$^{\text{NP}}$ (*to buy*)$^{\text{VP}}$)$^{\text{S}}$.

(c) Telephone volume was up sharply, but it was still at just half the level of the weekend (*preceding* ([Black **Monday**]$_{\text{Arg1}\Rightarrow\text{TMP}}$)$^{\text{NP}}$)$^{\text{VP}}$.

Figure 3
Examples of incorrect assignments by the meta-classifier. In each sentence, the verb is emphasized in italics and the head word for the selectional preferences is boldfaced. The argument under focus is marked within square brackets. $x \Rightarrow y$ means that the correct x label assigned by the baseline model is wrongly converted into y by the meta-classifier. As in Figure 2, examples also contain simplified syntactic annotations taken from the test set predicted syntactic layer.

examples are strong motivation for SP models that model both lexical and syntactic preferences. We will address such models in future work.

7. Conclusions

Current systems usually perform SRL in two pipelined steps: argument identification and argument classification. Whereas identification is mostly syntactic, classification requires semantic knowledge to be taken into account. In this article we have shown that the lexical heads seen in training data are too sparse to assign the correct role, and that selectional preferences are able to generalize those lexical heads. In fact, we show for the first time that the combination of the predictions of several selectional preference models with a state-of-the-art SRC system yields significant improvements in both in-domain and out-of-domain test sets. These improvements to role classification translate into small but statistically significant improvements in an end-to-end semantic role labeling system. We find these results encouraging considering that in the complete semantic role labeling task only a small percentage of argument candidates are affected by our modified role classification model. The experiments were carried out over the well-known CoNLL-2005 data set, based on PropBank.

We applied several selectional preference models, based on WordNet and distributional similarity. Our experiments show that all models outperform the pure lexical matching approach, with distributional methods performing better that WordNet-based methods, and second-order similarity models being the best. In addition to the traditional selectional preferences for verbs, we introduce the use of selectional preferences for prepositions, which are applied to classifying prepositional phrases. The combination of both types of selectional preferences improves over the use of selectional preferences for verbs alone.

The analysis performed over the cases where the base SRC system and the combined system differed showed that the selectional preferences are specially helpful when syntactic information is either incorrect or insufficient to disambiguate the correct role. The analysis also highlighted that the limitations of selectional preferences for modeling syntactic structures introduce some errors in the combined model. Those errors could be addressed if the SP models included some syntactic information.

Our research leaves the door open for tighter integration of semantic and syntactic information for Semantic Role Labeling. We introduced selectional preferences in the SRC system as simple features, but models which extend syntactic structures with

selectional preferences (or vice versa) could overcome some of the errors that our system introduced. Extending the use of selectional preferences to other syntactic types beyond noun phrases and prepositional phrases would be also of interest. In addition, the method for combining selectional preferences for verbs and prepositions was naive, and we expect that a joint model of verb and preposition preferences for prepositional phrases would improve results further. Finally, individual selectional preference methods could be improved and newer methods incorporated, which could further improve the results.

Acknowledgments

The authors would like to thank the three anonymous reviewers for their detailed and insightful comments on the submitted version of this manuscript, which helped us to improve it significantly in this revision. This work was partially funded by the Spanish Ministry of Science and Innovation through the projects OpenMT-2 (TIN2009-14675-C03) and KNOW2 (TIN2009-14715-C04-04). It also received financial support from the Seventh Framework Programme of the EU (FP7/2007- 2013) under grant agreements 247762 (FAUST) and 247914 (MOLTO). Mihai Surdeanu was supported by the Air Force Research Laboratory (AFRL) under prime contract no. FA8750-09-C-0181. Any opinions, findings, and conclusion or recommendations expressed in this material are those of the authors and do not necessarily reflect the view of the Air Force Research Laboratory (AFRL).

References

Agirre, Eneko, Timothy Baldwin, and David Martinez. 2008. Improving parsing and PP attachment performance with sense information. In *Proceedings of ACL-08: HLT*, pages 317–325, Columbus, OH.

Agirre, Eneko, Kepa Bengoetxea, Koldo Gojenola, and Joakim Nivre. 2011. Improving dependency parsing with semantic classes. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 699–703, Portland, OR.

Agirre, Eneko and David Martinez. 2001. Learning class-to-class selectional preferences. In *Proceedings of the 2001 Workshop on Computational Natural Language Learning (CoNLL-2001)*, pages 1–8, Toulouse.

Agirre, Eneko and German Rigau. 1996. Word sense disambiguation using conceptual density. In *Proceedings of the 16th Conference on Computational Linguistics - Volume 1*, COLING '96, pages 16–22, Stroudsburg, PA.

Baroni, Marco and Alessandro Lenci. 2010. Distributional memory: A general framework for corpus-based semantics. *Computational Linguistics*, 36(4):673–721.

Bergsma, Shane, Dekang Lin, and Randy Goebel. 2008. Discriminative learning of selectional preference from unlabeled text. In *Proceedings of EMNLP*, pages 59–68, Honolulu, HI.

Boas, H. C. 2002. Bilingual framenet dictionaries for machine translation. In *Proceedings of the Third International Conference on Language Resources and Evaluation (LREC)*, pages 1,364–1,371, Las Palmas de Gran Canaria.

Brockmann, Carsten and Mirella Lapata. 2003. Evaluating and combining approaches to selectional preference acquisition. In *Proceedings of the 10th Conference of the European Chapter of the Association of Computational Linguistics (EACL-2003)*, pages 27–34, Budapest.

Carreras, X. and L. Màrquez. 2004. Introduction to the CoNLL-2004 Shared Task: Semantic Role Labeling. In *Proceedings of the Eighth Conference on Computational Natural Language Learning (CoNLL-2004)*, pages 89–97, Boston, MA.

Carreras, X. and L. Màrquez. 2005. Introduction to the CoNLL-2005 Shared Task: Semantic Role Labeling. In *Proceedings of the Ninth Conference on Computational Natural Language Learning (CoNLL-2005)*, pages 152–164, Ann Arbor, MI.

Chakraborti, Sutanu, Nirmalie Wiratunga, Robert Lothian, and Stuart Watt. 2007. Acquiring word similarities with higher order association mining. In *Proceedings of the 7th International Conference on Case-Based Reasoning: Case-Based Reasoning Research and Development*, ICCBR '07, pages 61–76, Berlin.

Chambers, Nathanael and Daniel Jurafsky. 2010. Improving the use of pseudo-words for evaluating selectional preferences. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 445–453, Uppsala, Sweden.

Charniak, E. 2000. A maximum-entropy inspired parser. In *Proceedings of the 1st Meeting of the North American Chapter of the Association for Computational Linguistics (NAACL-2000)*, pages 132–139, Seattle, WA.

Clark, Stephen and Stephen Weir. 2002. Class-based probability estimation using a semantic hierarchy. *Computational Linguistics*, 28(2):187–206.

Edmonds, Philip. 1997. Choosing the word most typical in context using a lexical co-occurrence network. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics*, ACL '98, pages 507–509, Stroudsburg, PA.

Erk, Katrin. 2007. A simple, similarity-based model for selectional preferences. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics (ACL-2007)*, pages 216–223, Prague.

Erk, Katrin, Sebastian Padó, and Ulrike Padó. 2010. A flexible, corpus-driven model of regular and inverse selectional preferences. *Computational Linguistics*, 36(4):723–763.

Fillmore, C. J., J. Ruppenhofer, and C. F. Baker. 2004. FrameNet and representing the link between semantic and syntactic relations. In *Frontiers in Linguistics, volume I of Language and Linguistics Monograph Series B*. Institute of Linguistics, Academia Sinica, Taipei, pages 19–59.

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

Grefenstette, Gregory. 1992. Sextant: Exploring unexplored contexts for semantic extraction from syntactic analysis. In *ACL'92*, pages 324–326, Newark, DE.

Higashinaka, Ryuichiro and Hideki Isozaki. 2008. Corpus-based question answering for why-questions. In *Proceedings of the Third International Joint Conference on Natural Language Processing (IJCNLP)*, pages 418–425, Hyderabad.

Hindle, Donald. 1990. Noun classification from predicate-argument structures. In *Proceedings of the 28th Annual Meeting of the Association for Computational Linguistics (ACL-1990)*, pages 268–275, Pittsburgh, PA.

Koo, Terry, Xavier Carreras, and Michael Collins. 2008. Simple semi-supervised dependency parsing. In *Proceedings of ACL-08: HLT*, pages 595–603, Columbus, OH.

Lee, Lillian. 1999. Measures of distributional similarity. In *37th Annual Meeting of the Association for Computational Linguistics*, pages 25–32, College Park, MD.

Li, Hang and Naoki Abe. 1998. Generalizing case frames using a thesaurus and the MDL principle. *Computational Linguistics*, 24(2):217–244.

Lin, Dekang. 1998. Automatic retrieval and clustering of similar words. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and the 17th International Conference on Computational Linguistics (COLING-ACL-1998)*, pages 768–774, Montreal.

Litkowski, K. C. and O. Hargraves. 2005. The preposition project. In *Proceedings of the ACL-SIGSEM Workshop on the Linguistic Dimensions of Prepositions and their Use in Computational Linguistic Formalisms and Applications*, pages 171–179, Colchester.

Litkowski, K. C. and O. Hargraves. 2007. SemEval-2007 Task 06: Word-sense disambiguation of prepositions. In *Proceedings of the 4th International Workshop on Semantic Evaluations (SemEval-2007)*, pages 24–29, Prague.

Litkowski, Ken and Orin Hargraves. 2006. Coverage and inheritance in the preposition project. In *Prepositions '06: Proceedings of the Third ACL-SIGSEM Workshop on Prepositions*, pages 37–44, Trento.

Marcus, Mitchell, Grace Kim, Mary Ann Marcinkiewicz, Robert MacIntyre, Ann Bies, Mark Ferguson, Karen Katz, and Britta Schasberger. 1994. The Penn Treebank: Annotating predicate argument structure. In *Proceedings of the Workshop on Human Language Technology (HLT-94)*, pages 114–119, Plainsboro, NJ.

Màrquez, Lluís, Xavier Carreras, Kenneth C. Litkowski, and Suzanne Stevenson. 2008. Semantic role labeling: An introduction to the special issue. *Computational Linguistics*, 34(2):145–159.

McCarthy, Diana and John Carroll. 2003. Disambiguating nouns, verbs, and adjectives using automatically acquired selectional preferences. *Computational Linguisties*, 29:639–654.

Melli, Gabor, Yang Wang, Yudong Liu, Mehdi M. Kashani, Zhongmin Shi,

Baohua Gu, Anoop Sarkar, and Fred Popowich. 2005. Description of SQUASH, the SFU question answering summary handler for the DUC-2005 summarization task. In *Proceedings of Document Understanding Workshop, HLT/EMNLP Annual Meeting*, Vancouver.

Moschitti, Alessandro, Silvia Quarteroni, Roberto Basili, and Suresh Manandhar. 2007. Exploiting syntactic and shallow semantic kernels for question/answer classification. In *Proceedings of the 45th Conference of the Association for Computational Linguistics (ACL)*, pages 776–783, Prague.

Narayanan, S. and S. Harabagiu. 2004. Question answering based on semantic structures. In *Proceedings of the 20th International Conference on Computational Linguistics (COLING)*, pages 693–701, Geneva.

Noreen, E. W. 1989. *Computer-Intensive Methods for Testing Hypotheses: An Introduction*, Wiley.

Ó Séaghdha, Diarmuid. 2010. Latent variable models of selectional preference. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 435–444, Uppsala.

Padó, Sebastian and Mirella Lapata. 2007. Dependency-based construction of semantic space models. *Computational Linguistics*, 33(2):161–199.

Padó, Sebastian, Ulrike Padó, and Katrin Erk. 2007. Flexible, corpus-based modelling of human plausibility judgements. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL-2007)*, pages 400–409, Prague.

Palmer, M., D. Gildea, and P. Kingsbury. 2005. The proposition bank: An annotated corpus of semantic roles. *Computational Linguistics*, 31(1):71–105.

Pantel, Patrick, Rahul Bhagat, Bonaventura Coppola, Timothy Chklovski, and Eduard Hovy. 2007. ISP: Learning inferential selectional preferences. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, pages 564–571, Rochester, NY.

Pantel, Patrick and Dekang Lin. 2000. An unsupervised approach to prepositional phrase attachment using contextually similar words. In *Proceedings of the 38th Annual Conference of the Association of Computational Linguistics (ACL-2000)*, pages 101–108, Hong Kong.

Pradhan, S., W. Ward, and J. H. Martin. 2008. Towards robust semantic role labeling. *Computational Linguistics*, 34(2):289–310.

Rada, R., H. Mili, E. Bicknell, and M. Blettner. 1989. Development and application of a metric on semantic nets. *IEEE Transactions on Systems, Man, and Cybernetics*, 19(1):17–30.

Ratinov, Lev and Dan Roth. 2009. Design challenges and misconceptions in named entity recognition. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning (CoNLL-2009)*, pages 147–155, Boulder, CO.

Resnik, Philip. 1993a. *Selection and Information: A Class-Based Approach to Lexical Relationships*. Ph.D. thesis, University of Pennsylvania.

Resnik, Philip. 1993b. Semantic classes and syntactic ambiguity. In *Proceedings of the Workshop on Human Language Technology*, pages 278–283, Morristown, NJ.

Ritter, Alan, Mausam, and Oren Etzioni. 2010. A latent Dirichlet allocation method for selectional preferences. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 424–434, Uppsala.

Schütze, Hinrich. 1998. Automatic word sense discrimination. *Computational Linguistics*, 24(1):97–123.

Srikumar, V. and D. Roth. 2011. A joint model for extended semantic role labeling. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 129–139, Edinburgh.

Surdeanu, M., S. Harabagiu, J. Williams, and P. Aarseth. 2003. Using predicate-argument structures for information extraction. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics (ACL-2003)*, pages 8–15, Sapporo.

Surdeanu, Mihai, Massimiliano Ciaramita, and Hugo Zaragoza. 2011. Learning to rank answers to non-factoid questions from Web collections. *Computational Linguistics*, 37(2):351–383.

Surdeanu, Mihai, Lluís Màrquez, Xavier Carreras, and Pere R. Comas. 2007. Combination strategies for semantic role labeling. *Journal of Artificial Intelligence Research (JAIR)*, 29:105–151.

Sussna, Michael. 1993. Word sense disambiguation for free-text indexing using a massive semantic network. In *Proceedings of the Second International*

Conference on Information and Knowledge Management, CIKM '93, pages 67–74, New York, NY.

Wilks, Yorick. 1975. Preference semantics. In E. L. Kaenan, editor, *Formal Semantics of Natural Language*. Cambridge University Press, Cambridge, MA, pages 329–348.

Zapirain, Beñat, Eneko Agirre, and Lluís Màrquez. 2009. Generalizing over lexical features: Selectional preferences for semantic role classification. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the Association for Computational Linguistics and the 4th International Joint Conference on Natural Language Processing (ACL-IJCNLP-2009)*, pages 73–76, Suntec.

Zapirain, Beñat, Eneko Agirre, Lluís Màrquez, and Mihai Surdeanu. 2010. Improving semantic role classification with selectional preferences. In *Proceedings of the 11th Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL HLT 2010)*, pages 373–376, Los Angeles, CA.