Sequential SRL Using Selectional Preferences. An Approach with Maximum Entropy Markov Models

Beñat Zapirain, Eneko Agirre
IXA NLP Group
University of the Basque Country
Donostia, Basque Country
{benat.zapirain,e.agirre}@ehu.es

Lluís Màrquez
TALP Research Center
Technical University of Catalonia
Barcelona, Catalonia
lluism@lsi.upc.edu

Abstract

We present a sequential Semantic Role Labeling system that describes the tagging problem as a Maximum Entropy Markov Model. The system uses full syntactic information to select BIO-tokens from input data, and classifies them sequentially using state-of-the-art features, with the addition of Selectional Preference features. The system presented achieves competitive performance in the CoNLL-2005 shared task dataset and it ranks first in the SRL subtask of the Semeval-2007 task 17.

1 Introduction

In Semantic Role Labeling (SRL) the goal is to identify word sequences or arguments accompanying the predicate and assign them labels depending on their semantic relation. In this task we disambiguate argument structures in two ways: predicting VerbNet (Kipper et al., 2000) thematic roles and PropBank (Palmer et al., 2005) numbered arguments, as well as adjunct arguments.

In this paper we describe our system for the SRL subtask of the Semeval2007 task 17. It is based on the architecture and features of the system named 'model 2' of (Surdeanu et al., 2007), but it introduces two changes: we use Maximum Entropy for learning instead of AdaBoost and we enlarge the feature set with combined features and other semantic features.

Traditionally, most of the features used in SRL are extracted from automatically generated syntactic and lexical annotations. In this task, we also experiment with provided hand labeled semantic information for each verb occurrence such as the PropBank predicate sense and the Levin class. In addition, we use automatically learnt Selectional Preferences based on WordNet to generate a new kind of semantic based features.

We participated in both the "close" and the "open" tracks of Semeval2007 with the same system, making use, in the second case, of the larger CoNLL-2005 training set.

2 System Description

2.1 Data Representation

In order to make learning and labeling easier, we change the input data representation by navigating through provided syntactic structures and by extracting BIO-tokens from each of the propositions to be annotated as shown in (Surdeanu et al., 2007). These sequential tokens are selected by exploring the sentence spans or regions defined by the clause boundaries, and they are labeled with BIO tags depending on the location of the token: at the beginning, inside, or outside of a verb argument. After this data pre-processing step, we obtain a more compact and easier to process data representation, making also impossible overlapping and embedded argument predictions.

2.2 Feature Representation

Apart from Selectional Preferences (cf. Section 3) and those extracted from provided semantic information, most of the features we used are borrowed from the existing literature (Gildea and Jurafsky, 2002; Xue and Palmer, 2004; Surdeanu et al., 2007).

On the verb predicate:

- Form; Lemma; POS tag; Chunk type and Type of verb phrase; Verb voice; Binary flag indicat-

ing if the verb is a start/end of a clause.
- Subcategorization, i.e., the phrase structure rule expanding the verb parent node.
- VerbNet class of the verb (in the ”close” track only).

On the focus constituent:

- Type; Head;
- First and last words and POS tags of the constituent.
- POS sequence.
- Bag-of-words of nouns, adjectives, and adverbs in the constituent.
- TOP sequence: right-hand side of the rule expanding the constituent node; 2/3/4-grams of the TOP sequence.
- Governing category as described in (Gildea and Jurafsky, 2002).

Context of the focus constituent:

- Previous and following words and POS tags of the constituent.
- The same features characterizing focus constituents are extracted for the two previous and following tokens, provided they are inside the clause boundaries of the codified region.

Relation between predicate and constituent:

- Relative position; Distance in words and chunks; Level of embedding with respect to the constituent: in number of clauses.
- Binary position; if the argument is after or before the predicate.
- Constituent path as described in (Gildea and Jurafsky, 2002); All 3/4/5-grams of path constituents beginning at the verb predicate or ending at the constituent.
- Partial parsing path as described in (Carreras et al., 2004)); All 3/4/5-grams of path elements beginning at the verb predicate or ending at the constituent.
- Syntactic frame as described by Xue and Palmer (2004)

Combination Features

- Predicate and Phrase Type
- Predicate and binary position
- Head Word and Predicate
- Predicate and PropBank frame sense
- Predicate, PropBank frame sense, VerbNet class (in the ”close” track only)

2.3 Maximum Entropy Markov Models

Maximum Entropy Markov Models are a discriminative model for sequential tagging that models the local probability $P(s_n \mid s_{n-1}, o)$, where o is the context of the observation.

Given a MEMM, the most likely state sequence is the one that maximizes the following

$$S = argmax \prod_{i=1}^{n} P(s_i \mid s_{i-1}, o)$$

Translating the problem to SRL, we have role/argument labels connected to each state in the sequence (or proposition), and the observations are the features extracted in these points (token features). We get the most likely label sequence finding out the most likely state sequence (Viterbi).

All the conditional probabilities are given by the Maximum Entropy classifier with a tunable Gaussian prior from the Mallet Toolkit[1].

Some restrictions are considered when we search the most likely sequence[2]:

1. No duplicate argument classes for A0-A5 and thematic roles.
2. If there is a R-X argument (reference), then there has to be a X argument before (referenced).
3. If there is a C-X argument (continuation), then there has to be a X argument before.
4. Before a I-X token, there has to be a B-X or I-X token (because of the BIO encoding).
5. Given a predicate and its PropBank sense, only some arguments are allowed (e.g. not all the verbs support A2 argument).
6. Given a predicate and its Verbnet class, only some thematic roles are allowed.

3 Including Selectional Preferences

Selectional Preferences (SP) try to capture the fact that linguistic elements prefer arguments of a certain semantic class, e.g. a verb like ‘eat’ prefers as subject edible things, and as subject animate entities, as in “She was eating an apple” They can be learned from corpora, generalizing from the observed argument heads (e.g. ‘apple’, ‘biscuit’, etc.) into abstract classes (e.g. edible things). In our case we

[1] http://mallet.cs.umass.edu

[2] Restriction 5 applies to PropBank output. Restriction 6 applies to VerbNet output

follow (Agirre and Martinez, 2001) and use WordNet (Fellbaum, 1998) as the generalization classes (the concept `<food,nutrient>`).

The aim of using Selectional Preferences (SP) in SRL is to generalize from the argument heads in the training instances into general word classes. In theory, using word classes might overcome the data sparseness problem for the head-based features, but at the cost of introducing some noise.

More specifically, given a verb, we study the occurrences of the target verb in a training corpus (e.g. the PropBank corpus), and learn a set of SPs for each argument and adjunct of that verb. For instance, given the verb 'kill' we would have 2 SPs for each argument type, and 4 SPs for some of the observed adjuncts: `kill_A0`, `kill_A1`, `kill_AM-LOC`, `kill_AM-MNR`, `kill_AM-PNC` and `kill_AM-TMP`.

Rather than coding the SPs directly as features, we code the *predictions* instead, i.e. for each proposition in the training and testing set, we check the SPs for all the argument (and adjunct) headwords, and the SP which best fits the headword (see below) is the one that is selected. We codify the predicted argument (or adjunct) label as features, and we insert them among the corresponding argument features.

For instance, let's assume that the word 'railway' appears as the headword of a candidate argument of 'kill'. WordNet 1.6 yields the following hypernyms for 'railway' (from most general to most specific, we include the WordNet 1.6 concept numbers preceded by their specifity level);

```
1 00001740          1 00017954
2  00009457         2  05962976
3   00011937        3   05997592
4    03600463       4    06004580
5     03243979      5     06008236
6      03526208     6      06005839
7       03208595    7       02927599
                    8        03209020
```

Note that we do not care about the sense ambiguity and the explosion of concepts that it carries. Our algorithm will check each of the hypernyms of railway and match them with the concepts in the SPs of 'kill', giving preference to the most specific concept. In case that equally specific concepts match different SPs, we will choose the SP that has the concept that ranks highest in the SP, and code the SP feature with the label of the SP where the match succeeds. In the example, these are the most specific matches:

```
AM-LOC Con:03243979 Level:5 Ranking:32
A0     Con:06008236 Level:5 Ranking:209
```

There is a tie in the level, so we choose the one with the highest rank. All in all, this means that according to the learnt SPs we would predict that 'railway' is a location feature for 'kill', and we would therefore insert the 'SP:AM-LOC' feature among the argument features.

If 'railway' appears as the headword of other verbs, the predicted argument might be different. See for instance, the following verbs:

```
destroy:A1  Con:03243979 Level:5 Ranking:43
go:A0       Con:02927599 Level:7 Ranking:131
go:A2       Con:02927599 Level:7 Ranking:721
build:A1    Con:03209020 Level:8 Ranking:294
```

Note that our training examples did not contain 'railway' as an argument of any of these verbs, but due to the SPs we are able to code into a feature that 'railway' belongs to a concrete semantic class which contains conceptually similar headwords.

We decided to code the prediction of the SPs, rather than the SPs themselves, in order to be more robust to noise.

There is a further subtlety with our SP system. In order to label training and testing sets in similar conditions and avoid overfitting problems as much as possible, we split the training set into five folds and tagged each one with SPs learnt from the other four. For extracting SP features from test set examples, we use SPs learnt in the whole training set.

4 Experiments and Results

We participated in the "close" and the "open" tracks with the same classification model, but using different training sets in each one. In the close track we only use the provided training set, and in the open, the CoNLL-2005 training set (without VerbNet classes or thematic roles).

Before our participation, we tested the system in the CoNLL-2005 close track setting and it achieved competitive performance in comparison to the state-of-the-art results published in that challenge.

4.1 Semeval2007 setting

The data provided in the close track consists of the propositions of 50 different verb lemmas from PropBank (sections 02-21). The data for the CoNLL-2005 is also a subset of the PropBank data, but it

Track	Label	rank	prec.	rec.	F1
Close	VerbNet	1st	85.31	82.08	83.66
Close	PropBank	1st	85.04	82.07	83.52
Open	PropBank	1st	84.51	82.24	83.36

Table 1: Results in the SRL subtask of SemEval-2007 task 17

includes all the propositions in sections 02-21 and no VerbNet classes nor thematic roles for learning.

There is a total of 21 argument types for PropBank and 47 roles for VerbNet, which amounts to $21 * 2 + 1 = 43$ BIO-labels for PropBank predictions and $47 * 2 + 1 = 95$ for VerbNet. We filtered the less frequent (<5).

We trained the Maximum Entropy classifiers with 114,380 examples for the close track, and with 828,811 for the open track. We tuned the classifier by setting the Exponential Gaussian prior in 0.1

4.2 Results

In the close track we trained two classifiers, one to label PropBank numbered arguments and a second to label VerbNet thematic roles. Due to lack of time, we only trained the PropBank labels in the open track. Table 1 shows the results obtained in the SRL subtask. We ranked first in all of them, out of two participants.

4.3 Discussion

The results indicate that in the close track the system performs similarly on both PropBank arguments and VerbNet roles. The absence of VerbNet class-based features in the CoNLL-2005 training data could cause the loss of performance in the open track. We plan to perform the experiment on VerbNet roles for the open track to check the ability of the classifier to generalize across verbs.

Regarding the use of SP features, nowadays, we have not obtained relevant improvements in the predictions of the classifiers. It is our first approach to these kind of semantic features and there are more sophisticated but evident extraction variants which we are exploring.

Although the general performance is very similar without SP features, using them our system obtains better results in ARG3 core arguments and in the most frequent adjuncts such as location (*LOC*), general-purpose (*ADV*) and temporal (*TMP*).

We reproduced this improvements in experiments realized with CoNLL-2005 larger test sets. In that case, we improved ARG3-ARG4 core arguments as well as the mentioned adjuncts. There were more examples to be classified and we get better overall performance, but we need further experiments to be more conclusive.

5 Conclusions

We have presented a sequential semantic role labeling system for the Semeval-2007 task 17 (SRL). Based on Maximum Entropy Markov Models, it obtains competitive and promising results. We also have introduced semantic features extracted from Selectional Restrictions but we only have preliminary evidence of their usefulness.

Acknowledgements

We thank David Martinez for kindly providing the software that learnt the selectional preferences. This work has been partially funded by the Spanish education ministry (KNOW). Beñat is supported by a PhD grant from the University of the Basque Country.

References

E. Agirre and D. Martinez. 2001. Learning class-to-class selectional preferences. In *Proceedings of CoNLL-2001*, Toulouse, France.

X. Carreras, L. Màrquez, and G. Chrupala. 2004. Hierarchical recognition of propositional arguments with perceptrons. In *Proceedings of CoNLL-2004*.

C. Fellbaum. 1998. Wordnet: an electronic lexical database. *MIT press*.

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

K. Kipper, D. Hoa Trang, and M. Palmer. 2000. Class based construction of a verb lexicon. In *Proceedings of the 17th National Conference on Artificial Intelligence (AAAI-2000)*, Austin, TX, July.

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

M. Surdeanu, L. Màrquez, X. Carreras, and P. Comas. 2007. Combination strategies for semantic role labeling. *Journal of Artificial Intelligence Research (JAIR)*, 29:105–151.

N. Xue and M. Palmer. 2004. Calibrating features for semantic role labeling. In *Proceedings of EMNLP*.