

Semantic Parsing for High-Precision Semantic Role Labelling

Paola Merlo

Linguistics Department
University of Geneva
5 rue de Candolle
1211 Genève 4 Switzerland
merlo@lettres.unige.ch

Gabriele Musillo

Depts of Linguistics and Computer Science
University of Geneva
5 Rue de Candolle
1211 Genève 4 Switzerland
musillo4@etu.unige.ch

Abstract

In this paper, we report experiments that explore learning of syntactic and semantic representations. First, we extend a state-of-the-art statistical parser to produce a richly annotated tree that identifies and labels nodes with semantic role labels as well as syntactic labels. Secondly, we explore rule-based and learning techniques to extract predicate-argument structures from this enriched output. The learning method is competitive with previous single-system proposals for semantic role labelling, yields the best reported precision, and produces a rich output. In combination with other high recall systems it yields an F-measure of 81%.

1 Introduction

In statistical natural language processing, considerable ingenuity and insight have been devoted to developing models of syntactic information, such as statistical parsers and taggers. Successes in these syntactic tasks have recently paved the way to applying novel statistical learning techniques to levels of semantic representation, such as recovering the logical form of a sentence for information extraction and question-answering applications (Miller et al., 2000; Ge and Mooney, 2005; Zettlemoyer and Collins, 2007; Wong and Mooney, 2007).

In this paper, we also focus our interest on learning semantic information. Differently from other work that has focussed on logical form, however, we explore the problem of recovering the syntactic structure of the sentence, the propositional

argument-structure of its main predicates, and the substantive labels assigned to the arguments in the propositional structure, the semantic roles. This rich output can be useful for information extraction and question-answering, but also for anaphora resolution and other tasks for which the structural information provided by full syntactic parsing is necessary.

The task of semantic role labelling (SRL), as has been defined by previous researchers (Gildea and Jurafsky, 2002), requires collecting all the arguments that together with a verb form a predicate-argument structure. In most previous work, the task has been decomposed into the argument identification and argument labelling subtasks: first the arguments of each specific verb in the sentence are identified by classifying constituents in the sentence as arguments or not arguments. The arguments are then labelled in a second step.

We propose to produce the rich syntactic-semantic output in two steps, which are different from the argument identification and argument labelling subtasks. First, we generate trees that bear both syntactic and semantic annotation, such as those in Figure 1. The parse tree, however, does not explicitly encode information about predicate-argument structure, because it does not explicitly associate each semantic role to the verb that governs it. So, our second step consists in recovering the predicate-argument structure of each verb by gleaning this information in an already richly decorated tree.

There are linguistic and computational reasons to think that we can solve the joint problem of recovering the constituent structure of a sentence and its lexical semantics. From a linguistic point of view, the assumption that syntactic distributions will be predictive of semantic role assignments is based on linking theory (Levin, 1986). Linking theory assumes the existence of a hierarchy of se-

© 2008. Licensed under the *Creative Commons Attribution-Noncommercial-Share Alike 3.0 Unported* license (<http://creativecommons.org/licenses/by-nc-sa/3.0/>). Some rights reserved.

mantic roles which are mapped by default on a hierarchy of grammatical functions and syntactic positions, and it attempts to predict the mapping of the underlying semantic component of a predicate’s meaning onto the syntactic structure. For example, Agents are always mapped in syntactically higher positions than Themes. From a computational point of view, if the internal semantics of a predicate determines the syntactic expressions of constituents bearing a semantic role, it is then reasonable to expect that knowledge about semantic roles in a sentence will be informative of its syntactic structure. It follows rather naturally that semantic and syntactic parsing can be integrated into a single complex task.

Our proposal also addresses the problem of semantic role labelling from a slightly different perspective. We identify and label argument nodes first, while parsing, and we group them in a predicate-argument structure in a second step. Our experiments investigate some of the effects that result from organising the task of semantic role labelling in this way, and the usefulness of some novel features defined on syntactic trees.

In the remainder of the paper, we first illustrate the data and the graphical model that formalise the architecture used and its extension for semantic parsing. We then report on two kinds of experiments: we first evaluate the architecture on the joint task of syntactic and semantic parsing and then evaluate the joint approach on the task of semantic role labelling. We conclude with a discussion which highlights the practical and theoretical contribution of this work.

2 The Data

Our experiments on joint syntactic and semantic parsing use data that is produced automatically by merging the Penn Treebank (PTB) with PropBank (PRBK) (Marcus et al., 1993; Palmer et al., 2005), as shown in Figure 1. PropBank encodes propositional information by adding a layer of argument structure annotation to the syntactic structures of the Penn Treebank.¹ Verbal predicates in the Penn Treebank (PTB) receive a label REL and their arguments are annotated with abstract semantic role labels, such as A0, A1, or AA for those complements of the predicative verb that are considered arguments. Those complements of the verb la-

¹We use PRBK data as they appear in the CONLL 2005 shared task.

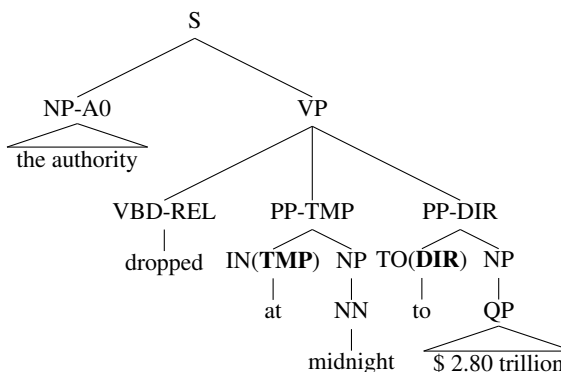


Figure 1: A sample syntactic structure with semantic role labels.

belled with a semantic functional label in the original PTB receive the composite semantic role label AM-*X*, where *X* stands for labels such as LOC, TMP or ADV, for locative, temporal and adverbial modifiers respectively. A tree structure with PropBank labels is shown in Figure 1. (The bold labels are not relevant for the moment and they will be explained later.)

3 The Syntactic and Semantic Parser Architecture

To achieve the complex task of joint syntactic and semantic parsing, we extend a current state-of-the-art statistical parser (Titov and Henderson, 2007) to learn semantic role annotation as well as syntactic structure. The parser uses a form of left-corner parsing strategy to map parse trees to sequences of derivation steps.

We choose this parser because it exhibits the best performance for a single generative parser, and does not impose hard independence assumptions. It is therefore promising for extensions to new tasks. Following (Titov and Henderson, 2007), we describe the original parsing architecture and our modifications to it as a Dynamic Bayesian network. Our description is brief and limited to the few aspects of interest here. For more detail, explanations and experiments see (Titov and Henderson, 2007). A Bayesian network is a directed acyclic graph that illustrates the statistical dependencies between the random variables describing a set of events (Jensen, 2001). Dynamic networks are Bayesian networks applied to unboundedly long sequences. They are an appropriate model for sequences of derivation steps in

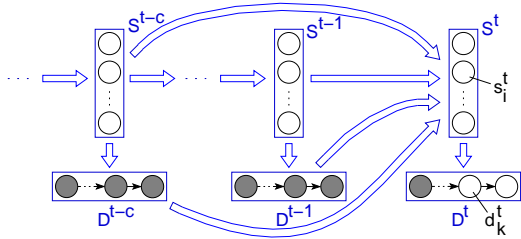


Figure 2: The pattern on connectivity and the latent vectors of variables in an Incremental Bayesian Network.

parsing (Titov and Henderson, 2007).

Figure 2 illustrates visually the main properties that are of relevance for us in this parsing architecture. Let T be a parse tree and D_1, \dots, D_m be the sequence of parsing decisions that has led to the building of this parse tree. Let also each parsing decision be composed of smaller parsing decisions d_1^1, \dots, d_k^1 , and let all these decisions be independent. Then,

$$\begin{aligned} P(T) &= P(D_1, \dots, D_m) \\ &= \prod_t P(D_t | D_1, \dots, D_{t-1}) \quad (1) \\ &= \prod_t \prod_k P(d_k^t | h(t, k)) \end{aligned}$$

where $h(t, k)$ denotes the parse history for sub-decision d_k^t .

The figure represents a small portion of the observed sequence of decisions that constitute the recovery of a parse tree, indicated by the observed states D_i . Specifically, it illustrates the pattern of connectivity for decision d_k^t . As can be seen the relationship between different probabilistic parsing decisions are not Markovian, nor do the decisions influence each other directly. Past decisions can influence the current decision through state vectors of independent latent variables, referred to as S_i . These state vectors encode the probability distributions of features of the history of parsing steps (the features are indicated by s_i^t in Figure 2).

As can be seen from the picture, the pattern of inter-connectivity allows previous non-adjacent states to influence future states. Not all states in the history are relevant, however. The inter-connectivity is defined dynamically based on the topological structure and the labels of the tree that is being developed. This inter-connectivity depends on a notion of structural locality (Henderson, 2003; Musillo and Merlo, 2006).²

²Specifically, the conditioning states are based on the

In order to extend this model to learn decisions concerning a joint syntactic-semantic representation, the semantic information needs to be highlighted in the model in several ways. We modify the network connectivity, and bias the learner.

First, we take advantage of the network’s dynamic connectivity to highlight the portion of the tree that bears semantic information. We augment the nodes that can influence parsing decisions at the current state by explicitly adding the vectors of latent variables related to the most recent child bearing a semantic role label of either type (REL, A0 to A5 or AM-X) to the connectivity of the current decision. These additions yield a model that is sensitive to regularities in structurally defined sequences of nodes bearing semantic role labels, within and across constituents. These extensions enlarge the locality domain over which dependencies between predicates bearing the REL label, arguments bearing an A0-A5 label, and adjuncts bearing an AM-X role can be specified, and capture both linear and hierarchical constraints between predicates, arguments and adjuncts. Enlarging the locality domain this way ensures for instance that the derivation of the role DIR in Figure 1 is not independent of the derivations of the roles TMP, REL (the predicate) and A0.

Second, this version of the Bayesian network tags its sentences internally. Following (Musillo and Merlo, 2005), we split some part-of-speech tags into tags marked with semantic role labels. The semantic role labels attached to a non-terminal directly projected by a preterminal and belonging to a few selected categories (DIR, EXT, LOC, MNR, PRP, CAUS or TMP) are propagated down to the pre-terminal part-of-speech tag of its head.³ This third extension biases the parser to learn the relationship between lexical items, semantic roles and the constituents in which they occur. This technique is illustrated by the bold labels in Figure 1.

We compare this augmented model to a simple baseline parser, that does not present any of the task-specific enhancements described above,

stack configuration of the left-corner parser and the derivation tree built so far. The nodes in the partially built tree and stack configuration that are selected to determine the relevant states are the following: *top*, the node on top of the pushdown stack before the current derivation move; the left-corner ancestor of *top* (that is, the second top-most node on the parser stack); the leftmost child of *top*; and the most recent child of *top*, if any.

³Exploratory data analysis indicates that these tags are the most useful to disambiguate parsing decisions.

	PTB/PRBK 24		
	P	R	F
Baseline	79.6	78.6	79.1
ST	80.5	79.4	79.9
ST+ EC	81.6	80.3	81.0

Table 1: Percentage F-measure (F), recall (R), and precision (P) of our joint syntactic and semantic parser on merged development PTB/PRBK data (section 24). Legend of models: ST=Split Tags; EC=enhanced connectivity.

other than being able to use the complex syntactic-semantic labels. Our augmented model has a total of 613 non-terminals to represent both the PTB and PropBank labels of constituents, instead of the 33 of the original syntactic parser. The 580 newly introduced labels consist of a standard PTB label followed by a set of one or more PropBank semantic role such as PP-AM-TMP or NP-A0-A1. As a result of lowering the six AM- X semantic role labels, 240 new part-of-speech tags were introduced to partition the original tag set which consisted of 45 tags. As already mentioned, argumental labels A0-A5 are specific to a given verb or a given verb sense, thus their distribution is highly variable. To reduce variability, we add the tag-verb pairs licensing these argumental labels to the vocabulary of our model. We reach a total of 4970 tag-word pairs. These pairs include, among others, all the tag-verb pairs occurring at least 10 times in the training data. In this very limited form of lexicalisation, all other words are considered unknown.

4 Parsing Experiments

Our extended joint syntactic and semantic parser was trained on sections 2-21 and validated on section 24 from the merged PTB/PropBank. To evaluate the joint syntactic and semantic parsing task, we compute the standard Parseval measures of labelled recall and precision of constituents, taking into account not only the original PTB labels, but also the newly introduced PropBank labels. This evaluation gives us an indication of how accurately and exhaustively we can recover this richer set of syntactic and semantic labels. The results, computed on the development data set from section 24 of the PTB with added PropBank annotation, are shown in Table 1. As the table indicates, both the enhancements based on semantic roles yield an im-

provement on the baseline.

This task enables us to compare, albeit indirectly, our integrated method to other methods where semantic role labels are learnt separately from syntactic structure. (Musillo and Merlo, 2006) report results of a merging technique where the output of the semantic role annotation produced by the best semantic role labellers in the 2005 CoNLL shared task is merged with the output of Charniak’s parser. Results range between 82.7% and 83.4% F-measure. Our integrated method almost reaches this level of performance.

The performance of the parser on the syntactic labels only (note reported in Table 1) is slightly degraded in comparison to the original syntax-only architecture (Henderson, 2003), which reported an F-measure of 89.1% since we reach 88.4% F-measure for the best syntactic-semantic model (last line of Table 1). This level of performance is still comparable to other syntactic parsers often used for extraction of semantic role features (88.2% F-measure) (Collins, 1999).

These results indicate that the extended parser is able to recover both syntactic and semantic labels in a fully connected parse tree. While it is true that the full fine-grained interpretation of the semantic label is verb-specific, the PropBank labels (A0,A1, etc) do respect some general trends. A0 labels are assigned to the most agentive of the arguments, while A1 labels tend to be assigned to arguments bearing a Theme role, and A2, A3, A4 and A5 labels are assigned to indirect object roles, while all the AM- X labels tend to be assigned to adjuncts. The fact that the parser learns these labels without explicit annotation of the link between the arguments and the predicate to which they are assigned, but based on the smoothed representation of the derivation of the parse tree and only very limited lexicalisation, appears to confirm linking theory, which assumes a correlation between the syntactic configuration of a sentence and the lexical semantic labels.

We need to show now that the quality of the output produced by the joint syntactic and semantic parsing is such that it can be used to perform other tasks where semantic role information is crucial. The most directly related task is semantic role labelling (SRL) as defined in the shared task of CoNLL 2005.

5 Extraction of Predicate-Argument Structures

Although there is reason to think that the good performance reported in the previous section is due to implicit learning of the relationship of the syntactic representation and the semantic role assignments, the output produced by the parser does not explicitly encode the predicate-argument structures. Collecting these associations is required to solve the semantic role labelling task as usually defined. We experimented with two methods: a simple rule-based method and a more complex learning method.

5.1 The rule-based method

The rule-based extraction method is the natural second step to solve the complete semantic role labelling task, after we identify and label semantic roles while parsing. Since in our proposal, we solve most of the problem in the first step, then we should be able to collect the predicate-argument pairs by simple, deterministic rules. The simplicity of the method also provides a useful comparison for more complex learning methods, which can be justified only if they perform better than simple rule-based predicate-argument extraction.

Our rule-based method automatically compiles finite-state automata defining the paths that connect the first node dominating a predicate to its semantic roles from parse trees enriched with semantic role labels.⁴ Such paths can then be used to traverse parse trees returned by the parsing model and collect argument structures. More specifically, a sample of sentences are randomly selected from the training section of the PTB/PRBK. For each predicate, then, all the arguments left and right of the predicate and all the adjuncts left and right respectively are collected and filtered by simple global constraints, thereby guaranteeing that only one type of obligatory argument label (A0 to A5) is assigned in each proposition.

When evaluated on gold data, this rule-based extraction method reaches 94.9% precision, 96.9% recall, for an F-measure of 95.9%. These results provide an upper bound as well as indicating that, while not perfect, the simple extraction rules reach a very good level of correctness if the input from the first step, syntactic and semantic parsing, is correct. The performance is much lower when

parses are not entirely correct, and semantic role labels are missing, as indicated by the results of 72.9% precision, 66.7% (F-measure 69.7%), obtained when using the best automatic parse tree. The fact that performance depends on the quality of the output of the first step, indicates that the extraction rules are sensitive to errors in the parse trees, as well as errors in the labelling. This indicates that a learning method might be more adapted to recover from these mistakes.

5.2 The SVM learning method

In a different approach to extract predicate argument structures from the parsing output, the second step learns to associate the right verb to each semantically annotated node (*srn*) in the tree produced in the first step. Each individual (*verb*, *srn*) pair in the tree is either a positive example (the *srn* is a member of the verb's argument structure) or a negative example (the argument either should not have been labelled as an argument or it is not associated to the verb). The training examples are produced by parsing section 2-21 of the merged PTB/PRBK data with the joint syntactic-semantic parser and producing the training examples by comparison with the CONLL 2005 gold propositions. There are approximately 800'000 training examples in total. These examples are used by an SVM classifier (Joachims, 1999).⁵ Once the predicate-argument structures are built, they are evaluated with the CONLL 2005 shared task criteria.

5.3 The learning features

The features used for the extraction of the predicate-argument structure reflect the syntactic properties that are useful to identify the arguments of a given verb. We use syntactic and semantic node label, the path between the verb and the argument, and the part-of-speech tag of the verb, which provides useful information about the tense of the verb. We also use novel features that encode *minimality conditions* and *locality constraints* (Rizzi, 1990). Minimality is a typical property of natural languages that is attested in several domains. In recovering predicate-argument structures, minimality guarantees that the arguments are related to the closest verb in a predicate domain, which is not always the verb to which they are connected by the

⁴It uses VanNoord's finite-state-toolkit <http://www.let.rug.nl/vannoord/Fsa/>.

⁵We use a radial basis function kernel, where parameters c and γ were determined by a grid search on a small subset of 2000 training examples. They are set at $c=8$ and $\gamma = 0.03125$.

shortest path. For example, the subject of an embedded clause can be closer to the verb of the main clause than to the predicate to which it should be attached. Minimality is encoded as a binary feature that indicates whether a verb w intervenes between the verb v and the candidate argument srn . Minimality is defined both in terms of linear precedence (indicated below as \prec) and of dominance within the same VP group. A VP group is a stack of VPs covering the same compound verb group, such as $[_{VP} \textit{should} [_{VP} \textit{have} [_{VP} [_{V} \textit{come}]]]]$. Formal definitions are given below:

$$\textit{minimal}(v, srn, w) =_{df}$$

$$\begin{cases} \textit{false} & \text{if } (v \prec w \prec srn \text{ or } srn \prec w \prec v) \text{ and} \\ & \text{VPG-dominates}(v, srn, w) \\ \textit{true} & \text{otherwise} \end{cases}$$

$$\textit{VPG-dominates}(v, srn, w) =_{df}$$

$$\begin{cases} \textit{true} & \text{if VP} \in \text{path}(v, srn) \text{ and} \\ & \text{VP} \in \text{VP-group directly dominating } w \\ \textit{false} & \text{otherwise} \end{cases}$$

In addition to the minimality conditions, which resolve ambiguity when two predicates compete to govern an argument, we use locality constraints to capture distinct local relationships between a verb and the syntactic position occupied by a candidate argument. In particular, we distinguish between internal arguments occupying a position dominated by a VP node, external arguments occupying a position dominated by an S node, and extracted arguments occupying a position dominated by an SBAR node. To approximate such structural distinctions, we introduce two binary features indicating, respectively, whether there is a node labelled S or SBAR on the path connecting the verb and the candidate argument.

6 Results and Discussion

Table 2 illustrates our results on semantic role labelling. Notice how much more precise the learning method is than the rule-based method, when the minimality constraint is added. The second and third line indicate that this contribution is mostly due to the minimality feature. The fifth and sixth line however illustrate that these features together are more useful than the widely used feature *path*. Recall however, suffers in the learnt method. Overall, the learnt method is better than a rule-based method only if path and either minimality or locality constraints are added, thus suggesting that

	Prec	Rec	F
Learning all features	87.4	63.6	73.7
Learning all –min	75.4	66.2	70.5
Learning all –loc	87.4	63.6	73.6
Rule-based	72.9	66.7	69.7
Learning all –path	80.6	60.9	69.4
Learning all –min –loc	74.3	63.8	68.6
Baseline	57.4	53.9	55.6

Table 2: Results on the development section (24), rule-based, and learning, (with all features, and without path, minimality and locality constraints) compared to a closest verb baseline.

the choice of features is crucial to reach a level of performance that justifies the added complexity of a learning method. Both methods are much better than a baseline that attaches each role to a verb by the shortest path.⁶ Notice that both these approaches are not lexicalised, they apply to all verbs. Learning experiments where the actual verbs were used showed a little degradation as well as a very considerable increase in training times (precision: 87.0%; recall: 61.0%; F: 71.7%).⁷

Some comments are in order to compare properly our best results – the learning method with all features – to other methods. Most of the best performing SRL systems are ensemble learners or rerankers, or they use external sources of information such as the PropBank frames files. While these techniques are effective to improve classification accuracy, we might want to compare the single systems, thus teasing apart the contribution of the features and the model from the contribution of the ensemble technique. Table 3 reports the single systems’ performance on the test set. These results seem to indicate that methods like ours, based on a first step of PropBank parsing, are comparable to other methods when learning regimes are factored out, contrary to pessimistic conclusions in previous work (Yi and Palmer, 2005). (Yi and Palmer, 2005) share the motivation of our work. They observe that the distributions of semantic la-

⁶In case of tie, the following verb is chosen for an A0 label and the preceding verb is chosen for all the other labels.

⁷We should notice that all these models encode the feature path as syntactic path, because in exploratory data analysis we found that this feature performed quite a bit better than path encoded taking into account the semantic roles assigned to the nodes on the path. Concerning the learning model, we notice that a simpler, and much faster to train, linear SVM classifier performs almost as well as the more complex RBF classifier. It is therefore preferable if speed is important.

Model	CONLL 23			Comments
	P	R	F	
(Surdeanu and Turmo, 2005)	80.3	73.0	76.5	Propbank frames to filter output, boosting
(Liu et al., 2005)	80.5	72.8	76.4	Single system + simple post-processing
(Moschitti et al., 2005)	76.6	75.2	75.9	Specialised kernels for each kind of role
This paper	87.6	65.8	75.1	Single system and model, locality features
(Ozgenicil and McCracken, 2005)	74.7	74.2	74.4	Simple system, no external knowledge
(Johansson and Nugues, 2005)	75.5	73.2	74.3	Uses only 3 sections for training

Table 3: Final Semantic Role Labelling results on test section 23 of Propbank as encoded in the CONLL shared task for those CONLL 2005 participants not using ensemble learning or external resources.

bels could potentially interact with the distributions of syntactic labels and redefine the boundaries of constituents, thus yielding trees that reflect generalisations over both these sources of information. They also attempt to assign SRL while parsing, by merging only the first two steps of the standard pipeline architecture, pruning and argument identification. Their parser outputs a binary argument-nonargument distinction. The actual fine-grained labelling is performed, as in other methods, by an ensemble classifier. Results are not among the best and Yi and Palmer conclude that PropBank parsing is too difficult and suffers from differences between chunk annotation and tree structure. We think instead that the method is promising, as shown by the results reported here, once the different factors that affect performance are teased apart.

Some qualitative observations on the errors are useful. On the one hand, as can be noticed in Table 3, our learning method yields the best precision, but often the worse recall and it has the most extreme difference between these two scores.⁸ This is very likely to be a consequence of the method. Since the assignment of the semantic role labels proper is performed during parsing, the number of nodes that require a semantic role is only 20% of the total. Therefore the parser develops a bias against assigning these roles in general, and recall suffers.⁹ On the other hand, precision is very good, thanks to the rich context in which the roles are assigned.

This property of our method suggests that combining our results with those of other existing se-

⁸This observation applies also in a comparison to the other systems that participated in the CONLL shared task.

⁹The SVM classifier, on the other hand, exceeds 94% in accuracy and its F measures are situated around 87–88% depending on the feature sets.

mantic role labellers might be beneficial, since the errors it performs are quite different. We tested this hypothesis by combining our outputs, which are the most precise, with the outputs of the system that reported the best recall (Haghighi et al., 2005). The combination, performed on sections 24 and 23, gives priority to our system when it outputs a non-null label (because of its high precision) and uses the other system’s label when our system outputs a null label. This combination produces a result of 79.0% precision, 80.4% recall, and 79.7% F-measure for section 24, and 80.5% precision, 81.4% recall, and 81.0% F-measure for section 23. We conclude that the combination is indeed able to exploit the positive aspects of both approaches, as the F-measure of the combined result is better than each individual result. It is also the best compared to the other systems of the CoNLL shared task. Comparatively, we find that applying the same combination technique to the output of the system by (Haghighi et al., 2005) with the output of the best system in the CoNLL 2005 shared task (Punyakonok et al., 2005) yields combined outputs that are not as good as the better of the two systems (P:76.3%; R:78.6%; F:77.4% for section 24; P:78.5%; R:80.0%; F:79.3% for section 23). This result confirms our initial hypothesis, that combination of systems with different performance characteristics yields greater improvement.

Another direct consequence of assigning roles in a rich context is that in collecting arguments for a given verb we hardly need to verify global constraints. Differently from previous work that had found that global coherence constraints considerably improved performance (Punyakonok et al., 2005), using global filtering constraints showed no improvement in our learning model. Thus, these results confirm the observations that a verb does

not assign its semantic roles independently of each other (Haghighi et al., 2005). Our method too can be seen as a way of formulating the SRL problem in a way that is not simply classification of each instance independently. Because identification of arguments and their labelling is done while parsing, the parsing history, both syntactic and semantic, is taken into account in identifying and labelling an argument. Semantic role labelling is integrated in structured sequence prediction. Further integration of semantic role labelling in structured probabilistic models related to the one described here has recently been shown to result in accurate synchronous parsers that derive both syntactic and semantic dependency representations (Henderson et al., 2008).

7 Conclusion

Overall our experiments indicate that an integrated approach to identification and labelling followed by predicate-argument recovery can solve the problem of semantic role labelling at a level of performance comparable to other approaches, as well as yielding a richly decorated syntactic-semantic parse tree. The high precision of our method yields very good results in combination with other high-recall systems. Its shortcomings indicates that future work lies in improving recall.

Acknowledgments

We thank the Swiss NSF for supporting this research under grant number 101411-105286/1, James Henderson for sharing the SSN software, and Xavier Carreras for providing the CoNLL-2005 data. Part of this work was completed while the second author was visiting MIT/CSAIL, hosted by Prof. Michael Collins.

References

Collins, Michael John. 1999. *Head-driven statistical models for natural language parsing*. Ph.D. thesis, University of Pennsylvania.

Ge, Ruiyang and Raymond J. Mooney. 2005. A statistical semantic parser that integrates syntax and semantics. In *Procs of CONLL-05*, Ann Arbor, Michigan.

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

Haghighi, Aria, Kristina Toutanova, and Christopher Manning. 2005. A joint model for semantic role labeling. In *Procs of CoNLL-2005*, pages 173–176, Ann Arbor, Michigan.

Henderson, Jamie. 2003. Inducing history representations for broad-coverage statistical parsing. In *Procs of NAACL-HLT'03*, pages 103–110, Edmonton, Canada.

Henderson, James, Paola Merlo, Gabriele Musillo and Ivan Titov. 2008. A latent variable model of synchronous parsing for syntactic and semantic dependencies. In *Procs of CoNLL'08 Shared Task*, Manchester, UK.

Jensen, Finn V. 2001. *Bayesian networks and decision graphs*. Springer Verlag.

Joachims, Thorsten. 1999. Making large-scale svm learning practical. In Schlkopf, B., C. Burges, and A. Smola, editors, *Advances in Kernel Methods - Support Vector Learning*. MIT Press.

Johansson, Richard and Pierre Nugues. 2005. Sparse bayesian classification of predicate arguments. In *Procs of CoNLL-2005*, pages 177–180, Ann Arbor, Michigan.

Levin, Lori. 1986. *Operations on lexical form: unaccusative rules in Germanic languages*. Ph.D. thesis, Massachusetts Institute of Technology.

Liu, Ting, Wanxiang Che, Sheng Li, Yuxuan Hu, and Huaijun Liu. 2005. Semantic role labeling system using maximum entropy classifier. In *Procs of CoNLL-2005*, pages 189–192, Ann Arbor, Michigan.

Marcus, Mitch, Beatrice Santorini, and M.A. Marcinkiewicz. 1993. Building a large annotated corpus of English: the Penn Treebank. *Computational Linguistics*, 19:313–330.

Miller, S., H. Fox, L. Ramshaw, and R. Weischedel. 2000. A novel use of statistical parsing to extract information from text. In *Procs of NAACL 2000*.

Moschitti, Alessandro, Ana-Maria Giuglea, Bonaventura Coppola, and Roberto Basili. 2005. Hierarchical semantic role labeling. In *Procs of CoNLL-2005*, pages 201–204, Ann Arbor, Michigan.

Musillo, Gabriele and Paola Merlo. 2005. Lexical and structural biases for function parsing. In *Procs of IWPT'05*, pages 83–92, Vancouver, British Columbia, October.

Musillo, Gabriele and Paola Merlo. 2006. Accurate semantic parsing of the proposition bank. In *Procs of NAACL'06*, New York, NY.

Ozgenicil, Necati Ercan and Nancy McCracken. 2005. Semantic role labeling using libSVM. In *Procs of CoNLL-2005*, pages 205–208, Ann Arbor, Michigan, June.

Palmer, Martha, Daniel Gildea, and Paul Kingsbury. 2005. The Proposition Bank: An annotated corpus of semantic roles. *Computational Linguistics*, 31:71–105.

Punyakanok, Vasin, Peter Koomen, Dan Roth, and Wen tau Yih. 2005. Generalized inference with multiple semantic role labeling systems. In *Procs of CoNLL-2005*, Ann Arbor, MI USA.

Rizzi, Luigi. 1990. *Relativized minimality*. MIT Press, Cambridge, MA.

Surdeanu, Mihai and Jordi Turmo. 2005. Semantic role labeling using complete syntactic analysis. In *Procs of CoNLL'05*, Ann Arbor, Michigan.

Titov, Ivan and James Henderson. 2007. Constituent parsing with Incremental Sigmoid Belief Networks. In *Procs of ACL'07*, pages 632–639, Prague, Czech Republic.

Wong, Yuk Wah and Raymond Mooney. 2007. Learning synchronous grammars for semantic parsing with lambda calculus. In *Procs of ACL'07*, pages 960–967, Prague, Czech Republic.

Yi, Szu-ting and Martha Palmer. 2005. The integration of semantic parsing and semantic role labelling. In *Procs of CoNLL'05*, Ann Arbor, Michigan.

Zettlemoyer, Luke and Michael Collins. 2007. Online learning of relaxed CCG grammars for parsing to logical form. In *Procs of EMNLP-CoNLL'07*, pages 678–687.