Semantic Role Labeling via FrameNet, VerbNet and PropBank

Ana-Maria Giuglea and **Alessandro Moschitti**
Department of Computer Science
University of Rome "Tor Vergata"
Rome, Italy
agiuglea@gmail.com
moschitti@info.uniroma2.it

Abstract

This article describes a robust semantic parser that uses a broad knowledge base created by interconnecting three major resources: FrameNet, VerbNet and PropBank. The FrameNet corpus contains the examples annotated with semantic roles whereas the VerbNet lexicon provides the knowledge about the syntactic behavior of the verbs. We connect VerbNet and FrameNet by mapping the FrameNet frames to the VerbNet Intersective Levin classes. The PropBank corpus, which is tightly connected to the VerbNet lexicon, is used to increase the verb coverage and also to test the effectiveness of our approach. The results indicate that our model is an interesting step towards the design of more robust semantic parsers.

1 Introduction

During the last years a noticeable effort has been devoted to the design of lexical resources that can provide the training ground for automatic semantic role labelers. Unfortunately, most of the systems developed until now are confined to the scope of the resource used for training. A very recent example in this sense was provided by the CONLL 2005 shared task (Carreras and Màrquez, 2005) on PropBank (PB) (Kingsbury and Palmer, 2002) role labeling. The systems that participated in the task were trained on the Wall Street Journal corpus (WSJ) and tested on portions of WSJ and Brown corpora. While the best F-measure recorded on WSJ was 80%, on the Brown corpus, the F-measure dropped below 70%. The most significant causes for this performance decay

were highly ambiguous and unseen predicates (i.e. predicates that do not have training examples).

The same problem was again highlighted by the results obtained with and without the frame information in the Senseval-3 competition (Litkowski, 2004) of FrameNet (Johnson et al., 2003) role labeling task. When such information is not used by the systems, the performance decreases by 10 percent points. This is quite intuitive as the semantics of many roles strongly depends on the focused frame. Thus, we cannot expect a good performance on new domains in which this information is not available.

A solution to this problem is the automatic frame detection. Unfortunately, our preliminary experiments showed that given a FrameNet (FN) predicate-argument structure, the task of identifying the associated frame can be performed with very good results when the verb predicates have enough training examples, but becomes very challenging otherwise. The predicates belonging to new application domains (i.e. not yet included in FN) are especially problematic since there is no training data available.

Therefore, we should rely on a semantic context alternative to the frame (Giuglea and Moschitti, 2004). Such context should have a wide coverage and should be easily derivable from FN data. A very good candidate seems to be the Intersective Levin class (ILC) (Dang et al., 1998) that can be found as well in other predicate resources like PB and VerbNet (VN) (Kipper et al., 2000).

In this paper we have investigated the above claim by designing a semi-automatic algorithm that assigns ILCs to FN verb predicates and by carrying out several semantic role labeling (SRL) experiments in which we replace the frame with the ILC information. We used support vector ma-

chines (Vapnik, 1995) with (a) polynomial kernels to learn the semantic role classification and (b) Tree Kernels (Moschitti, 2004) for learning both frame and ILC classification. Tree kernels were applied to the syntactic trees that encode the subcategorization structures of verbs. This means that, although FN contains three types of predicates (nouns, adjectives and verbs), we only concentrated on the verb predicates and their roles.

The results show that: (1) ILC can be derived with high accuracy for both FN and Probank and (2) ILC can replace the frame feature with almost no loss in the accuracy of the SRL systems. At the same time, ILC provides better predicate coverage as it can also be learned from other corpora (e.g. PB).

In the remainder of this paper, Section 2 summarizes previous work done on FN automatic role detection. It also explains in more detail why models based exclusively on this corpus are not suitable for free-text parsing. Section 3 focuses on VN and PB and how they can enhance the robustness of our semantic parser. Section 4 describes the mapping between frames and ILCs whereas Section 5 presents the experiments that support our thesis. Finally, Section 6 summarizes the conclusions.

2 Automatic Semantic Role Labeling

One of the goals of the FN project is to design a linguistic ontology that can be used for the automatic processing of semantic information. The associated hierarchy contains an extensive semantic analysis of verbs, nouns, adjectives and situations in which they are used, called *frames*. The basic assumption on which the frames are built is that each word evokes a particular situation with specific participants (Fillmore, 1968). The word that evokes a particular frame is called *target word* or *predicate* and can be an adjective, noun or verb. The participant entities are defined using semantic roles and they are called *frame elements*.

Several models have been developed for the automatic detection of the frame elements based on the FN corpus (Gildea and Jurafsky, 2002; Thompson et al., 2003; Litkowski, 2004). While the algorithms used vary, almost all the previous studies divide the task into: 1) the identification of the verb arguments to be labeled and 2) the tagging of each argument with a role. Also, most of the models agree on the core features as be-

ing: *Predicate*, *Headword*, *Phrase Type*, *Governing Category*, *Position*, *Voice* and *Path*. These are the initial features adopted by Gildea and Jurafsky (2002) (henceforth G&J) for both frame element identification and role classification.

One difference among previous machine-learning models is whether they used the frame information or not. The impact of the frame feature over unseen predicates and words is particularly interesting for us. The results obtained by G&J provide some interesting insights in this direction. In one of their experiments, they used the frame to generalize from predicates seen in the training data to unseen predicates, which belonged to the same frame. The overall performance increased showing that when no training data is available for a target word we can use data from the same frame.

Other studies suggest that the frame is crucial when trying to eliminate the major sources of errors. In their error analysis, (Thompson et al., 2003) pinpoints that the verb arguments with headwords that are rare in a particular frame but not rare over the whole corpus are especially hard to classify. For these cases the frame is very important because it provides the context information needed to distinguish between different word senses.

Overall, the experiments presented in G&J's study correlated with the results obtained in the Senseval-3 competition show that the frame feature increases the performance and decreases the amount of annotated examples needed in training (i.e. frame usage improves the generalization ability of the learning algorithm). On the other hand, the results obtained without the frame information are very poor.

These results show that having broader frame coverage is very important for robust semantic parsing. Unfortunately, the 321 frames that contain at least one verb predicate cover only a small fraction of the English verb lexicon and of the possible domains. Also from these 321 frames only 100 were considered to have enough training data and were used in Senseval-3 (see (Litkowski, 2004) for more details).

Our approach for solving such problems involves the usage of a frame-like feature, namely the Intersective Levin class (ILC). We show that the ILC can replace the frame with almost no loss in performance. At the same time, ILC provides better coverage as it can be learned also from other

corpora (e.g. PB).

The next section provides the theoretical support for the unified usage of FN, VN and PB, explaining why and how it is possible to link them.

3   Linking FrameNet to VerbNet and PropBank

In general, predicates belonging to the same FN frame have a coherent syntactic behavior that is also different from predicates pertaining to other frames (G&J). This finding is consistent with theories of linking that claim that the syntactic behavior of a verb can be predicted from its semantics (Levin, 1993). This insight justifies the attempt to use ILCs instead of the frame feature when classifying FN semantic roles (Giuglea and Moschitti, 2004).

The main advantage of using Levin classes comes from the fact that other resources like PB and the VN lexicon contain this kind of information. Thus, we can train an ILC classifier also on the PB corpus, considerably increasing the verb knowledge base at our disposal. Another advantage derives from the syntactic criteria that were applied in defining the Levin's clusters. As shown later in this article, the syntactic nature of these classes makes them easier to classify than frames when using only syntactic and lexical features.

More precisely, Levin's clusters are formed according to diathesis alternation criteria which are variations in the way verbal arguments are grammatically expressed when a specific semantic phenomenon arises. For example, two different types of diathesis alternations are the following:

(a) Middle Alternation

[$_{Subject}$, $_{Agent}$ The butcher] cuts [$_{Direct\ Object}$, $_{Patient}$ **the meat**].
[$_{Subject}$, $_{Patient}$ **The meat**] cuts easily.

(b) Causative/inchoative Alternation

[$_{Subject}$, $_{Agent}$ Janet] broke [$_{Direct\ Object}$, $_{Patient}$ **the cup**].
[$_{Subject}$, $_{Patient}$ **The cup**] broke.

In both cases, what is alternating is the grammatical function that the Patient role takes when changing from the transitive use of the verb to the intransitive one. The semantic phenomenon accompanying these types of alternations is the change of focus from the entity performing the action to the theme of the event.

Levin documented 79 alternations which constitute the building blocks for the verb classes. Although alternations are chosen as the primary means for identifying the classes, additional properties related to subcategorization, morphology and extended meanings of verbs are taken into account as well. Thus, from a syntactic point of view, the verbs in one Levin class have a regular behavior, different from the verbs pertaining to other classes. Also, the classes are semantically coherent and all verbs belonging to one class share the same participant roles.

This constraint of having the same semantic roles is further ensured inside the VN lexicon which is constructed based on a more refined version of the Levin's classification, called Intersective Levin classes (ILCs) (Dang et al., 1998). The lexicon provides a regular association between the syntactic and semantic properties of each of the described classes. It also provides information about the syntactic frames (alternations) in which the verbs participate and the set of possible semantic roles.

One corpus associated with the VN lexicon is PB. The annotation scheme of PB ensures that the verbs belonging to the same Levin class share similarly labeled arguments. Inside one ILC, to one argument corresponds one semantic role numbered sequentially from ARG0 to ARG5. The adjunct roles are labeled ARGM.

Levin classes were constructed based on regularities exhibited at grammatical level and the resulting clusters were shown to be semantically coherent. As opposed, the FN frames were built on semantic bases, by putting together verbs, nouns and adjectives that evoke the same situations. Although different in conception, the FN verb clusters and VN verb clusters have common properties[1]:

1. Different syntactic properties between distinct verb clusters (as proven by the experiments in G&J)

2. A shared set of possible semantic roles for all verbs pertaining to the same cluster.

Having these insights, we have assigned a correspondent VN class not to each verb predicate but rather to each frame. In doing this we have applied the simplifying assumption that a frame has a

---

[1]See section 4.4 for more details

unique corresponding Levin class. Thus, we have created a one-to-many mapping between the ILCs and the frames. In order to create a pair ⟨FN frame, VN class⟩, our mapping algorithm checks both the syntactic and semantic consistency by comparing the role frequency distributions on different syntactic positions for the two candidates. The algorithm is described in detail in the next section.

4 Mapping FrameNet frames to VerbNet classes

The mapping algorithm consists of three steps: (a) we link the frames and ILCs that have the largest number of verbs in common and we create a set of pairs ⟨FN frame, VN class⟩ (see Table 1); (b) we refine the pairs obtained in the previous step based on diathesis alternation criteria, i.e. the verbs pertaining to the FN frame have to undergo the same diathesis alternation that characterize the corresponding VN class (see Table 2) and (c) we manually check the resulting mapping.

4.1 The mapping algorithm

Given a frame, F, we choose as candidate for the mapping the ILC, C, that has the largest number of verbs in common with it (see Table 1, line (I)). If the number is greater or equal than three we form a pair ⟨F, C⟩ that will be tested in the second step of the algorithm. Only the frames that have more than 3 verb lexical units are candidates for this step (frames with less than 3 members cannot pass condition (II)). This excludes a number of 60 frames that will be subsequently manually mapped.

In order to assign a VN class to a frame, we have to verify that the verbs belonging to the FN frame participate in the same diathesis alternation criteria used to define the VN class. Thus, the pairs ⟨F, C⟩ formed in step 1 of the mapping algorithm have to undergo a validation step that verifies the similarity between the enclosed FN frame and VN class. This validation process has several sub-steps:

First, we make use of the property (2) of the Levin classes and FN frames presented in the previous section. According to this property, all verbs pertaining to one frame or ILC have the same participant roles. Thus, a first test of compatibility between a frame and a Levin class is that they share the same participant roles. As FN is annotated with frame-specific semantic roles, we manually mapped these roles into the VN set of the-

INPUT
$VN = \{C | C \text{ is a } VerbNet \text{ class}\}$
$VN \text{ Class } C = \{v | c \text{ is a verb of } C\}$
$FN = \{F | F \text{ is a } FrameNet \text{ frame}\}$
$FN \text{ frame } F = \{v | v \text{ is a verb of } F\}$
OUTPUT
$Pairs = \{\langle F, C \rangle | F \in FN, C \in VN : F \text{ maps to } C\}$
COMPUTE PAIRS:
$Let \ Pairs = \emptyset$
$for \ each \ F \in FN$
  $(I) \ compute \ C^* = arg \max_{C \in VN} |F \cap C|$
  $(II) \ if \ |F \cap C^*| \geq 3 \ then \ Pairs = Pairs \cup \langle F, C^* \rangle$

Table 1: Linking FrameNet frames and VerbNet classes.

$TR = \{\theta i : \theta i \text{ is the } i - th \text{ theta role of VerbNet }\}$

$for \ each \ \langle F, C \rangle \in Pairs$
$\overrightarrow{A}^F = \langle o_1, .., o_n \rangle, o_i = \#\langle \theta_i, F, pos = \text{adjacent} \rangle$
$\overrightarrow{D}^F = \langle o_1, .., o_n \rangle, o_i = \#\langle \theta_i, F, pos = \text{distant} \rangle$
$\overrightarrow{A}^C = \langle o_1, .., o_n \rangle, o_i = \#\langle \theta_i, C, pos = \text{adjacent} \rangle$
$\overrightarrow{D}^C = \langle o_1, .., o_n \rangle, o_i = \#\langle \theta_i, C, pos = \text{distant} \rangle$

$Score_{F,C} = \frac{2}{3} \times \frac{\overrightarrow{A}^F \cdot \overrightarrow{A}^C}{\left\|\overrightarrow{A}^F\right\| \times \left\|\overrightarrow{A}^C\right\|} + \frac{1}{3} \times \frac{\overrightarrow{D}^F \cdot \overrightarrow{D}^C}{\left\|\overrightarrow{D}^F\right\| \times \left\|\overrightarrow{D}^C\right\|}$

Table 2: Mapping algorithm - refining step.

matic roles. Given a frame, we assigned thematic roles to all frame elements that are associated with verbal predicates. For example the *Speaker*, *Addressee*, *Message* and *Topic* roles from the *Telling* frame were respectively mapped into the *Agent*, *Recipient*, *Theme* and *Topic* theta roles.

Second, we build a frequency distribution of VN thematic roles on different syntactic positions. Based on our observation and previous studies (Merlo and Stevenson, 2001), we assume that each ILC has a distinct frequency distribution of roles on different grammatical slots. As we do not have matching grammatical functions in FN and VN, we approximate that *subjects* and *direct objects* are more likely to appear on positions adjacent to the predicate, while *indirect objects* appear on more distant positions. The same intuition is successfully used by G&J to design the *Position* feature.

For each thematic role θ_i we acquired from VN and FN data the frequencies with which θ_i appears on an adjacent A or distant D positions in a given frame or VN class (i.e. #⟨θ_i, class, position⟩). Therefore, for each frame and class, we obtain two vectors with thematic role frequencies corresponding respectively to the adjacent and distant positions (see Table 2). We compute a score for each

Score	No. of Frames	Not mapped	Correct	Overall Correct
[0,0.5]	118	48.3%	82.5%	
(0.5,0.75]	69	0	84%	89.6%
(0.75,1]	72	0	100%	

Table 3: Results of the mapping algorithm.

pair $\langle F, C \rangle$ using the normalized scalar product.

The core arguments, which tend to occupy adjacent positions, show a minor syntactic variability and are more reliable than adjunct roles. To account for this in the overall score, we multiply the adjacent and the distant scores by 2/3 and 1/3, respectively. This limits the impact of adjunct roles like Temporal and Location.

The above frequency vectors are computed for FN directly from the corpus of predicate-argument structure examples associated with each frame. The examples associated with the VN lexicon are extracted from the PB corpus. In order to do this we apply a preprocessing step in which each label Arg0..5 is replaced with its corresponding thematic role given the ILC of the predicate. We assign the same roles to the adjuncts all over PB as they are general for all verb classes. The only exception is ARGM-DIR that can correspond to Source, Goal or Path. We assign different roles to this adjunct based on the prepositions. We ignore some adjuncts like ARGM-ADV or ARGM-DIS because they cannot bear a thematic role.

4.2 Mapping Results

We found that only 133 VN classes have correspondents among FN frames. Moreover, from the frames mapped with an automatic score smaller than 0.5 almost a half did not match any of the existing VN classes[2]. A summary of the results is depicted in Table 3. The first column contains the automatic score provided by the mapping algorithm when comparing frames with ILCs. The second column contains the number of frames for each score interval. The third column contains the percentage of frames that did not have a corresponding VN class and finally the fourth and fifth columns contain the accuracy of the mapping algorithm for each interval score and for the whole task, respectively.

We mention that there are 3,672 distinct verb senses in PB and 2,351 distinct verb senses in

---

[2]The automatic mapping is improved by manually assigning the FN frames of the pairs that receive a score lower than 0.5.

FN. Only 501 verb senses are in common between the two corpora which means 13.64% of PB and 21.31% of FN. Thus, by training an ILC classifier on both PB and FN we extend the number of available verb senses to 5,522.

4.3 Discussion

In the literature, other studies compared the Levin classes with the FN frames, e.g. (Baker and Ruppenhofer, 2002; Giuglea and Moschitti, 2004; Shi and Mihalcea, 2005). Their findings suggest that although the two set of clusters are roughly equivalent there are also several types of mismatches:

1. Levin classes that are narrower than the corresponding frames,

2. Levin classes that are broader that the corresponding frames and

3. Overlapping groups.

For our task, point 2 does not pose a problem. Points 1 and 3 however suggest that there are cases in which to one FN frame corresponds more than one Levin class. By investigating such cases, we noted that the mapping algorithm consistently assigns scores below 75% to cases that match problem 1 (two Levin classes inside one frame) and below 50% to cases that match problem 3 (more than two Levin classes inside one frame). Thus, to increase the accuracy of our results, a first step should be to assign independently an ILC to each of the verbs pertaining to frames with score lower than 0.75%.

Nevertheless the current results are encouraging as they show that the algorithm is achieving its purpose by successfully detecting syntactic incoherences that can be subsequently corrected manually. Also, in the next section we will show that our current mapping achieves very good results, giving evidence for the effectiveness of the Levin class feature.

5 Experiments

In the previous sections we have presented the algorithm for annotating the verb predicates of FrameNet (FN) with Intersective Levin classes (ILCs). In order to show the effectiveness of this annotation and of the ILCs in general we have performed several experiments.

First, we trained (1) an ILC multiclassifier from FN, (2) an ILC multiclassifier from PB and (3) a

	Run 51.3.2	Cooking 45.3	Characterize 29.2	Other_cos 45.4	Say 37.7	Correspond 36.1	Multiclassifier
PB #Train Instances	262	6	2,945	2,207	9,707	259	52,172
PB #Test Instances	5	5	134	149	608	20	2,742
PB Results	**75**	**33.33**	**96.3**	**97.24**	**100**	**88.89**	**92.96**
FN #Train Instances	5,381	138	765	721	1,860	557	46,734
FN #Test Instances	1,343	35	40	184	1,343	111	11,650
FN Results	**96.36**	**72.73**	**95.73**	**92.43**	**94.43**	**78.23**	**92.63**

Table 4: F1s of some individual ILC classifiers and the overall multiclassifier accuracy (180 classes on PB and 133 on FN).

	Body_part	Crime	Degree	Agent	Multiclassifier
FN #Train Instances	1,511	39	765	6,441	102,724
FN #Test Instances	356	5	187	1,643	25,615
LF+Gold Frame	90.91	88.89	70.51	93.87	90.8
LF+Gold ILC	90.80	88.89	71.52	92.01	88.23
LF+Automatic Frame	**84.87**	**88.89**	**70.10**	**87.73**	**85.64**
LF+Automatic ILC	**85.08**	**88.89**	**69.62**	**87.74**	**84.45**
LF	79.76	75.00	64.17	80.82	80.99

Table 5: F1s of some individual FN role classifiers and the overall multiclassifier accuracy (454 roles).

frame multiclassifier from FN. We compared the results obtained when trying to classify the VN class with the results obtained when classifying frame. We show that ILCs are easier to detect than FN frames.

Our second set of experiments regards the automatic labeling of FN semantic roles on FN corpus when using as features: gold frame, gold ILC, automatically detected frame and automatically detected ILC. We show that in all situations in which the VN class feature is used, the accuracy loss, compared to the usage of the frame feature, is negligible. This suggests that the ILC can successfully replace the frame feature for the task of semantic role labeling.

Another set of experiments regards the generalization property of the ILC. We show the impact of this feature when very few training data is available and its evolution when adding more and more training examples. We again perform the experiments for: gold frame, gold ILC, automatically detected frame and automatically detected ILC.

Finally, we simulate the difficulty of free text by annotating PB with FN semantic roles. We used PB because it covers a different set of verbal predicates and also because it is very different from FN at the level of vocabulary and sometimes even syntax. These characteristics make PB a difficult testbed for the semantic role models trained on FN.

In the following section we present the results obtained for each of the experiments mentioned above.

5.1 Experimental setup

The corpora available for the experiments were PB and FN. PB contains about 54,900 predicates and gold parse trees. We used sections from 02 to 22 (52,172 predicates) to train the ILC classifiers and Section 23 (2,742 predicates) for testing purposes. The number of ILCs is 180 in PB and 133 on FN, i.e. the classes that we were able to map.

For the experiments on FN corpus, we extracted 58,384 sentences from the 319 frames that contain at least one verb annotation. There are 128,339 argument instances of 454 semantic roles. In our evaluation we use only verbal predicates. Moreover, as there is no fixed split between training and testing, we randomly selected 20% of sentences for testing and 80% for training. The sentences were processed using Charniak's parser (Charniak, 2000) to generate parse trees automatically.

The classification models were implemented by means of the SVM-light-TK software available at http://ai-nlp.info.uniroma2.it/moschitti which encodes tree kernels in the SVM-light software (Joachims, 1999). We used the default parameters. The classification performance was evaluated using the F_1 measure for the individual role and ILC classifiers and the accuracy for the multiclassifiers.

5.2 Automatic VerbNet class vs. automatic FrameNet frame detection

In these experiments, we classify ILCs on PB and frames on FN. For the training stage we use SVMs with Tree Kernels.

The main idea of tree kernels is the modeling of a $K_T(T_1,T_2)$ function which computes the number of common substructures between two trees T_1 and T_2. Thus, we can train SVMs with structures drawn directly from the syntactic parse tree of the sentence. The kernel that we employed in our experiments is based on the SCF structure devised in (Moschitti, 2004). We slightly modified SCF by adding the headwords of the arguments, useful for representing the selectional preferences (more details are given in (Giuglea and Moschitti, 2006).

For frame detection on FN, we trained our classifier on 46,734 training instances and tested on 11,650 testing instances, obtaining an accuracy of 91.11%. For ILC detection the results are depicted in Table 4. The first six columns report the F_1 measure of some verb class classifiers whereas the last column shows the global multiclassifier accuracy. We note that ILC detection is more accurate than the frame detection on both FN and PB. Additionally, the ILC results on PB are similar with those obtained for the ILCs on FN. This suggests that the training corpus does not have a major influence. Also, the SCF-based tree kernel seems to be robust in what concerns the quality of the parse trees. The performance decay is very small on FN that uses automatic parse trees with respect to PB that contains gold parse trees.

5.3 Automatic semantic role labeling on FrameNet

In the experiments involving semantic role labeling, we used SVMs with polynomial kernels. We adopted the standard features developed for semantic role detection by Gildea and Jurafsky (see Section 2). Also, we considered some of the features designed by (Pradhan et al., 2005): *First and Last Word/POS in Constituent*, *Subcategorization*, *Head Word of Prepositional Phrases* and the *Syntactic Frame* feature from (Xue and Palmer, 2004). For the rest of the paper, we will refer to these features as being literature features (LF). The results obtained when using the literature features alone or in conjunction with the gold frame feature, gold ILC, automatically detected frame feature and automatically detected ILC are depicted in Table 5.
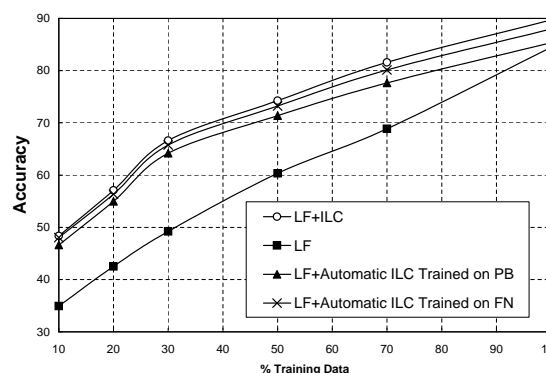

Figure 1: Semantic role learning curve.

The first four columns report the F_1 measure of some role classifiers whereas the last column shows the global multiclassifier accuracy. The first row contains the number of training and testing instances and each of the other rows contains the performance obtained for different feature combinations. The results are reported for the labeling task as the argument-boundary detection task is not affected by the frame-like features (G&J).

We note that automatic frame produces an accuracy very close to the one obtained with automatic ILC suggesting that this is a very good candidate for replacing the frame feature. Also, both automatic features are very effective and they decrease the error rate by 20%.

To test the impact of ILC on SRL with different amount of training data, we additionally draw the learning curves with respect to different features: LF, LF+ (gold) ILC, LF+automatic ILC trained on PB and LF+automatic ILC trained on FN. As can be noted, the automatic ILC information provided by the ILC classifiers (trained on FN or PB) performs almost as good as the gold ILC.

5.4 Annotating PB with FN semantic roles

To show that our approach can be suitable for semantic role free-text annotation, we have automatically classified PB sentences[3] with the FN semantic-role classifiers. In order to measure the quality of the annotation, we randomly selected 100 sentences and manually verified them. We measured the performance obtained with and without the automatic ILC feature. The sentences contained 189 arguments from which 35 were incorrect when ILC was used compared to 72 incorrect in the absence of this feature, i.e. an accuracy of 81% with ILC versus 62% without it. This demonstrates the importance of the ILC feature

---

[3]The results reported are only for role classification.

outside the scope of FN where the frame feature is not available.

6 Conclusions

In this paper we have shown that the ILC feature can successfully replace the FN frame feature. By doing that we could interconnect FN to VN and PB obtaining better verb coverage and a more robust semantic parser. Our good results show that we have defined an effective framework which is a promising step toward the design of more robust semantic parsers.

In the future, we intend to measure the effectiveness of our system by testing FN SRL on a larger portion of PB or on other corpora containing a larger verb set.

References

Collin Baker and Josef Ruppenhofer. 2002. Framenets frames vs. levins verb classes. In *28th Annual Meeting of the Berkeley Linguistics Society*.

Xavier Carreras and Lluís Màrquez. 2005. Introduction to the CoNLL-2005 shared task: Semantic role labeling. In *Proceedings of CoNLL-2005*.

Eugene Charniak. 2000. A maximum-entropy-inspired parser. In *Proceedings of NACL00*, Seattle, Washington.

Hoa Trang Dang, Karin Kipper, Martha Palmer, and Joseph Rosenzweig. 1998. Investigating regular sense extensions based on intersective levin classes. In *Coling-ACL98*.

Charles J. Fillmore. 1968. The case for case. In *Universals in Linguistic Theory*.

Daniel Gildea and Daniel Jurafsky. 2002. Automatic labeling of semantic roles. *Computational Linguistic*.

Ana-Maria Giuglea and Alessandro Moschitti. 2004. Knowledge discovering using FrameNet, VerbNet and PropBank. In *Proceedings of Workshop on Ontology and Knowledge Discovering at ECML 2004*, Pisa, Italy.

Ana-Maria Giuglea and Alessandro Moschitti. 2006. Shallow semantic parsing based on FrameNet, VerbNet and PropBank. In *Proceedings of the 17th European Conference on Artificial Intelligence*, Riva del Garda, Italy.

T. Joachims. 1999. Making large-scale SVM learning practical. In B. Schölkopf, C. Burges, and A. Smola, editors, *Advances in Kernel Methods - Support Vector Learning*.

Christopher Johnson, Miriam Petruck, Collin Baker, Michael Ellsworth, Josef Ruppenhofer, and Charles Fillmore. 2003. Framenet: Theory and practice. *Berkeley, California*.

Paul Kingsbury and Martha Palmer. 2002. From Treebank to PropBank. In *LREC02)*.

Karin Kipper, Hoa Trang Dang, and Martha Palmer. 2000. Class-based construction of a verb lexicon. In *AAAI00*.

Beth Levin. 1993. *English Verb Classes and Alternations A Preliminary Investigation*. Chicago: University of Chicago Press.

Kenneth Litkowski. 2004. Senseval-3 task automatic labeling of semantic roles. In *Senseval-3*.

Paola Merlo and Suzanne Stevenson. 2001. Automatic verb classification based on statistical distribution of argument structure. *CL Journal*.

Alessandro Moschitti. 2004. A study on convolution kernels for shallow semantic parsing. In *ACL04*, Barcelona, Spain.

Sameer Pradhan, Kadri Hacioglu, Valeri Krugler, Wayne Ward, James H. Martin, and Daniel Jurafsky. 2005. Support vector learning for semantic argument classification. *Machine Learning Journal*.

Lei Shi and Rada Mihalcea. 2005. Putting pieces together: Combining FrameNet, VerbNet and WordNet for robust semantic parsing. In *Proceedings of Cicling 2005, Mexico*.

Cynthia A. Thompson, Roger Levy, and Christopher Manning. 2003. A generative model for semantic role labeling. In *14th European Conference on Machine Learning*.

V. Vapnik. 1995. *The Nature of Statistical Learning Theory*. Springer.

Nianwen Xue and Martha Palmer. 2004. Calibrating features for semantic role labeling. In *Proceedings of EMNLP 2004*, Barcelona, Spain. Association for Computational Linguistics.