Knowledge Discovering using FrameNet, VerbNet and PropBank

Ana-Maria Giuglea[1] and Alessandro Moschitti[2]

[1] University of Texas at Dallas, Richardson, TX 75083-0688, USA
ana-maria.giuglea@student.utdallas.edu
[2] University of Rome Tor Vergata, Computer Science Department
00133 Roma (Italy),
moschitti@info.uniroma2.it

Abstract. In this paper, we present a high accurate system for FrameNet semantic role classification based on the innovative features derived from a combined use of FrameNet, VerbNet and PropBank. The main property of our approach is a unified view of the above three resources which is theoretically supported by the linking theory. Experiments on Support Vector Machines (SVM) show that our system classifies semantic information with high accuracy, enabling future work for the design of knowledge discovering FrameNet-based models.

1 Introduction

One of the aims the FrameNet project (`www.icsi.berkeley .edu/~framenet`) relates to the design of a linguistic ontology hierarchy useful to automatically derive and process new information. This hierarchy contains an extensive semantic analysis of verbs, nouns and adjectives and their case-frame representations. The basic assumption on which the frames are built is that each word evokes a particular situation with particular participants. The situations can be fairly simple depicting the entities involved and the roles they serve or can be very complex and in this case they are called scenarios. A scenario usually implies a set of assumptions and practices; all entities and events that are developed inside are to be understood in connection to the specific rules imposed by the frame. The participant entities are called semantic roles and the word that evokes a particular frame is called target word or predicate. The target word can be also thought as a function that describes the relation among different entities taking part into an event and for this reason the participants are often seen as predicate's arguments. At sentence level, a target word together with its dependent semantic roles form a predicate-argument structure. There are many levels of representation for the information captured by a predicate-argument structure; the FrameNet ontology captures predicate-argument structures at semantic level.

This semantic information can be used by knowledge management systems to perform a semantic analysis similar to the one that the Semantic Web community proposes. The remarkable difference is that semantic roles can be assigned with high accuracy automatically. For example the following sentences have been annotated according to the Arrest frame. They contain semantic roles like, *Offence*, *Suspect* and *Authorities* that are specific to this frame:

(a) [$_{Time}$ One Saturday night in the summer of 1966] [$_{Authorities}$ police in Brooklyn] [$_{Target}$ apprehended] [$_{Suspect}$ sixteen teenagers of both sexes aged between sixteen and nineteen who were dancing naked in the street].

(b) [$_{Suspect}$ Devlin] was [$_{Target}$ apprehended] [$_{Offence}$ in the course of robbing a jeweler's shop in Fernley Shopping Centre].

An Internet query trying to retrieve information about **teenagers arrested** will return all documents in which a teenager appears in the context of an arrest even if he is not the offender or worst, even if he has nothing in common with the felony (example a). The same happens if we try to find out **who was arrested for robbing the jeweler in Fernley Shopping Centre** (example b). This case is even more complicated as the complexity of the query increases with its length. If we have the semantic role information available we can look for a *Suspect* that committed a specific *Offence* in a particular situation (frame). This results in a noticeable increase of the accuracy of the answer.

Given the importance of the frame information, several machine learning models have been developed, e.g. [1, 2] to derive FrameNet semantic roles automatically. Other work, e.g. [3] focus on the extraction of predicate argument structures as they are defined in PropBank [4]. The annotation of PropBank is based on the Levin's verb classes defined in the VerbNet lexicon [5]. In VerbNet the arguments of the verb are represented at semantic level and thus they have associated semantic roles. As a consequence VerbNet lexicon can act as a liaison between PropBank syntactic arguments and FrameNet semantic arguments. To our knowledge no approach for predicate argument extraction uses together the above three resources.

In this article, we describe an algorithm based on a linguistic model that is used to identify, in a sentence, semantic roles that are related to a verb target word. We base our system on the unique combination of VerbNet, PropBank and FrameNet and on a state-of-the-art learning algorithm that uses Support Vector Machines (SVMs). To prove the benefit of our approach we implemented also the standard literature model to carry out a comparative analysis.

2 From PropBank to FrameNet via VerbNet

The linguistic work that describes the interaction between syntax and semantics is very extensive and known under the common name of linking theory. There are approaches that advocate the direct mapping of semantic structures into the surface syntactic form (mono-stratal frameworks) and other that use an intermediate grammatical level to facilitate the transition (multi-stratal frameworks). We chose the latter as the problem of automatically detecting predicate-argument structures at grammatical level seems to be much easier than actually detecting semantic roles directly as it has been shown in [3, 2].

Semantic roles are defined relating to the grammatical level such that they provide a generalization over the uses of a grammatical argument in a specific semantic frame. As a consequence semantic roles (or participant roles) can be considered as labels that link grammatical aspects of an argument to the role it plays in the situation evoked by the target word.

The minimum number of the arguments of a verb is zero, like for the weather verbs (e.g. *rain* and *snow*) but there are examples in which a verb can take up to four arguments or more (e.g. *John leased the apartment to Bill for 1000$ a months.*). Recognition of such argument-structures is made even more difficult by the multiple ways in which the same event having the same participant roles can be realized at syntactic level. For example the following sentences have very different syntactic structures but share the same meaning and the same semantic roles: A met B; A and B met; a meeting between A and B took place; A had a meeting with B; A and B had a meeting.

Our final goal is to correctly identify the participants in the event no matter how they were syntactically expressed. In order to achieve this task is more feasible to concentrate first on the simpler problem of detecting grammatical level arguments and use this intermediate level to link to the semantic level.

2.1 Semantic Roles vs. PropBank Arguments

When speaking about the theory of argument structure and the linking between syntax and semantics the question that arises more often is how much of the semantic meaning can be inferred based on the syntactic behavior. One of the most comprehensive studies on the subject that discusses the case of the verbs is [6]. Levin builds a verb classification starting from the assumption that there is a strong connection between syntax and semantics. Verbs are grouped together based on their syntactic behavior and the resulting clusters are coherent from a semantic point of view as all verbs in one Levin class share the same semantic roles. The Levin clusters are formed at grammatical level according to diathesis alternation criteria. Diathesis alternations are defined as being variations in the way verbal-arguments are grammatically expressed consistently with a specific semantic phenomenon. For example two different types of diathesis alternation are the following:

(a) Middle Alternation
[$_{Subject,\ Arg0,\ Agent}$ The butcher] cuts [$_{Direct\ Object,\ Arg1,\ Patient}$ **the meat**].
[$_{Subject,\ Arg1,\ Patient}$ **The meat**] cuts easily.

(b) Causative/inchoative Alternation
[$_{Subject,\ Arg0,\ Agent}$ Janet] broke [$_{Direct\ Object,\ Arg1,\ Patient}$ **the cup**].
[$_{Subject,\ Arg1,\ Patient}$ **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. More precisely the semantic role of the subject of the intransitive use of the verb is the same as the semantic role of the direct object of the transitive use. 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.

In the examples above we used three levels of representation: grammatical level, diathesis level and semantic level. In PropBank the arguments are annotated at a diathesis level, i.e. for the verbs pertaining to the same Levin class that participate in the same diathesis alternations the arguments will be the same. Also, as can be noted from the example, inside the same verb class one PropBank argument corresponds to a single semantic role.

In PropBank, predicates exhibiting the same diathesis alternation share similarly-labeled arguments provided that the verbs belong to the same semantic verb class defined in VerbNet. No attempt is made to ensure consistency of mapping between argument labels and the semantic roles played by the arguments unless the predicates belong to the same semantic verb class. The senses of one verb are defined at a coarse-grained level according to the classes in which the verb is listed as member.

2.2 Linking between PropBank, VerbNet and FrameNet

PropBank is [4] a 300.000-word corpus of Wall Street Journal articles tagged with predicate-argument relations. The annotation on this corpus is based on the Levin's verb classification. The expected arguments of each Levin's sense are numbered sequentially from Arg0 to Arg5. Higher numbered argument labels are less consistent and assigned per-verb basis.

In general for lower numbered arguments some regularity can be observed. For example, subjects of transitive verbs are assigned the label Arg0, while Arg1 corresponds to the role of the direct object. Arg0 can appear in special cases on the syntactic position of an object. One of these cases is the class of induced action verbs when the grammatical subject is not the one performing the action but rather the entity causing it. For these cases the label assigned to the grammatical subject is ArgA. Arg0, Arg1..ArgA are called core arguments and are specific for each semantic frame of one verb. There are other arguments that are independent of the semantic frame (e.g. temporal or location denoting arguments) called adjuncts.

As we mentioned before one property of the PropBank annotation is that predicates exhibiting the same diathesis alternations share similarly-labeled arguments provided that the verbs belong to the same Levin verb class. Inside one Levin class to one argument corresponds only one semantic role. As VerbNet is also constructed on top of the same verb classification it follows the restriction. Given the above assumptions, in order to obtain the semantic level in predicate argument structures we need both pieces of information: the verb class and the type of the PropBank argument. To obtain the grammatical level arguments we developed a system that was trained on PropBank [2] and that automatically annotated this information. For acquiring the verb class information we used the VerbNet lexicon.

As our final goal is to get access to the information contained in the FrameNet Ontology we design a mapping between the VerbNet classes and the FrameNet frames. For example into the VerbNet class *Judgment* we mapped the FrameNet frames: *Rewards and punishments*, *Judgment communication*, *Sentencing*, *Notification of charges*, *Arraignment*, *Court examination*, *Pardon*, *Try defendant*, *Forgiveness*, *Jury deliberation* and *Judgment direct address.*

As VerbNet is based on intersective Levin classes in order to perform a joining with FrameNet frames we have to follow the principle that assigns a verb to a class only if it shares the same diathesis alternations with the other verbs of that class. We started by collapsing together the VerbNet classes and FrameNet frames following a simple heuristic that combined the classes that have the most verbs in common. The second phase was to refine the joining applying diathesis alternation criteria. This process is semi-automated as the output of this algorithm is manually corrected.

Given the above mapping and the automatic PropBank argument extractor (in general more accurate than the semantic role parser), we extended the standard features [1] used for semantic role classification with: (a) the *PropBank Arguments* feature, i.e. FrameNet is automatically annotated with the arguments from PropBank, (b) the *Verb Class* feature, i.e. the Levin's verb class associated with the predicate word and (3) the *Diathesis Alternation* feature, i.e. the whole predicate argument structures. This is the sequence of all PropBank arguments associated with the target verb.

3 The Experiments

To prove the benefit of our new features we compared the semantic role classification performance using the standard and the extended features. For each feature set we report two different performances: (1) the single argument classifiers and (2) the multi-classifier, i.e. the final combination of all the argument classifiers.

The corpora available for the experiments were: PropBank (`www.cis.upenn.edu/∼ace`) along with Penn TreeBank 2 [7] and FrameNet. PropBank contains about 53,700 sentences and a fixed split between training and testing which has been used in other researches, e.g. [3]. For the FrameNet corpus we extracted 52,395 sentences from the 319 frames that contain at least one verb annotation for a total of 120,951 arguments for verbs. Only verbs are selected to be predicates in our evaluations. Moreover, as there is no fixed split between training and testing, we selected randomly 20% of sentences for testing and 80% for training. The sentences were processed using Collins' parser [8] to generate parse-trees automatically.

The classification performance was evaluated using the f_1 measure[3] for single arguments and the accuracy for the final multi-class classifier. This latter choice allows us to compare the results with previous literature works, e.g. [1, 3]. The experiments were carried out using the SVM-light software [9] available at `svmlight.joachims.org` with the default polynomial kernel of degree[4] = 3.

Table 1 shows the individual results for 5 roles out of 482 verbs' total roles whereas the multi-classifier accuracy refers to all 482 roles. Row 2 reports the number of testing instances for the selected semantic roles. Row 3 shows the f_1 measures for the individual semantic role classifiers, trained using standard features only. Row 4 illustrates the f_1 performances for the individual classifiers trained with the standard and the extended features. In Multi-Classifier column are shown the multi-classifier performance using the standard and extended features.

[3] f_1 assigns equal importance to Precision P and Recall R, i.e. $f_1 = \frac{2P \cdot R}{P+R}$.

[4] In [2] it has been shown that the best performing degree for both FrameNet and PropBank is 3.

Table 1. f_1 and accuracy of the argument classifiers and the overall multi-classifier for FrameNet semantic roles.

Semantic Role (f_1 measure)	Agent	Theme	Degree	Goal	Instrument	Manner	Multi-Classifier Accuracy
Testing Instances	174	850	53	542	5	163	
Standard Features	92.0	90.3	74.9	85.9	67.9	81.0	85.2
Extended Features	94.4	98.4	84.5	94.3	88.9	80.0	90.9

We note that the extended features improve argument type classification (using standard features) of about 6 absolute percent points (90.9 vs. 85.2). This is an important result as: (a) it allows the classification accuracy of semantic roles to reach 91 % and (b) it confirms the validity of the Levin approach in providing semantic classification on syntactic bases. In fact, the syntactic alternation, described by the PropBank argument sequence, restricts the number of possible Levin classes suitable for the target verb. In turn, the verb classes suggest the allowed senses for the target verb. This helps the classifier to derive much easily the correct set of semantic roles.

References

1. Gildea, D., Jurasfky, D.: Automatic labeling of semantic roles. Computational Linguistic **28** (2002) 496–530
2. Moschitti, A.: A study on convolution kernel for shallow semantic parsing. To appear in the Proceedings of ACL-2004, Barcelona, Spain (2004)
3. Pradhan, S., Hacioglu, K., Krugler, V., Ward, W., Martin, J.H., Jurafsky, D.: Support vector learning for semantic argument classification. to appear in Journal of Machine Learning Research (2004)
4. Kingsbury, P., Palmer, M.: From Treebank to PropBank. In: Proceedings of LREC-02),. (Las Palmas, Spain, 2002)
5. Kipper, K., Dang, H.T., Palmer, M.: Class-based construction of a verb lexicon. In: Proceedings of AAAI-2000 Seventeenth National Conference on Artificial Intelligence, Austin, TX. (2000)
6. Levin, B.: English Verb Classes and Alternations A Preliminary Investigation. Chicago: University of Chicago Press. (1993)
7. Marcus, M.P., Santorini, B., Marcinkiewicz, M.A.: Building a large annotated corpus of english: The Penn Treebank. Computational Linguistics **19** (1993)
8. Collins, M.: Three generative, lexicalized models for statistical parsing. In: Proceedings of the ACL and EACL, Somerset, New Jersey (1997) 16–23
9. Joachims, T.: Making large-scale SVM learning practical. In Schölkopf, B., Burges, C., Smola, A., eds.: Advances in Kernel Methods - Support Vector Learning. (1999)