UTD-SRL: A Pipeline Architecture for Extracting Frame Semantic Structures

Cosmin Adrian Bejan and **Chris Hathaway**
Human Language Technology Research Institute
The University of Texas at Dallas
Richardson, TX 75083-0688, USA
{ady,chris}@hlt.utdallas.edu

Abstract

This paper describes our system for the task of extracting frame semantic structures in SemEval–2007. The system architecture uses two types of learning models in each part of the task: Support Vector Machines (SVM) and Maximum Entropy (ME). Designed as a pipeline of classifiers, the semantic parsing system obtained competitive precision scores on the test data.

1 Introduction

The SemEval–2007 task for extracting frame semantic structures relies on the human annotated data available in the FrameNet (FN) database. The Berkeley FrameNet project (Baker et al., 1998) is an ongoing effort of building a semantic lexicon for English based on the theory of *frame semantics*. In frame semantics, the meaning of words or word expressions, also called *target words* (TW), comprises aspects of conceptual structures, or *frames*, that describe specific situations. The semantic roles, or *frame elements* (FE), associated with a target word are locally defined in the frame evoked by the target word. Currently, the FN lexicon includes more than 135,000 sentences extracted from the British National Corpus containing more than 6,100 target words that evoke more than 825 semantic frames.

For this task, we extended our previous work at Senseval-3 (Bejan et al., 2004) by (1) experimenting with additional features, (2) adding new classification sub-tasks to accomplish all the requirements, and (3) integrating these sub-tasks into a pipeline architecture.

2 System Description

Given a sentence, the frame semantic structure extraction task consists of recognizing the word expressions that evoke semantic frames, assigning the correct frame to them and, for each target word, detecting and labeling the corresponding frame elements properly. The task also requires the determination of syntactic realizations associated to a frame element, such as *grammatical function* (GF) and *phrase type* (PT). The following illustrates a sentence example annotated with frame elements together with their corresponding grammatical functions and phrase types for the target word *"tie"*:

Frame = Make_Cognitive_Connection
evokes

AEOI's activities and facilities	have been	tied	to several universities	.
FE = Content1 GF = Ext PT = NP			FE = Content2 GF = Dep PT = PP	

To extract semantic structures similar to those illustrated in the example we divide the SemEval–2007 task into four sub-tasks: (1) target word frame disambiguation (TWFD); (2) FE boundary detection (FEBD); (3) GF label classification (GFLC) and (4) FE label classification (FELC). The sub-tasks TWFD and GFLC are natural extensions of the approach described in (Bejan et al., 2004) for the task of semantic role labeling at Senseval-03. We design machine learning classifiers specific for each of the four sub-tasks and arrange them in a pipeline architecture such that a classifier can use information predicted by its previous classifiers. The system architecture is illustrated in Figure 1. In the data processing step, we parse each sentence into a syntactic tree using the Collins parser and extract named entities using an in

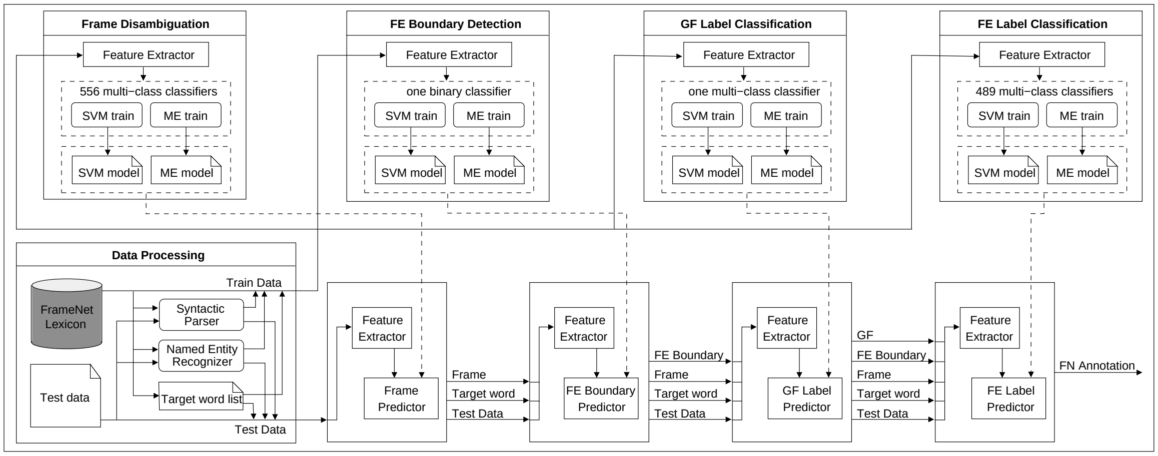

Figure 1: System architecture.

house implementation of a named entity recognizer. We also extract from the FN lexicon mappings of target words and the semantic frames they evoke.

Various features corresponding to constituents were extracted and passed to SVM and ME classifiers. For example, in Figure 2, the frame dis-

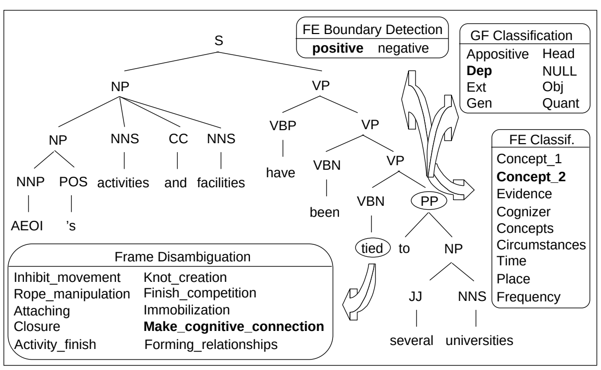

Figure 2: Classification examples for each sub-task.

ambiguation sub-task extracts features corresponding to the constituent *tied* in order to predict the right frame between the semantic frames that can be evoked by this target word. In this figure, the correct categories for each sub-task are shown in boldface.

The complete set of features extracted for all the classification sub-tasks is illustrated in Figure 3. These represent a subset of features used in previous works (Gildea and Jurafsky, 2002; Florian et al., 2002; Surdeanu et al., 2003; Xue and Palmer, 2004; Bejan et al., 2004; Pradhan et al., 2005) for automatic semantic role labeling and word sense disambiguation. Figure 3 also indicates whether or not a feature is selected for a specific classification task.

In the remaining part of this section we describe in detail each classification sub-task and the features that have the most salient effect on improving the corresponding classifiers.

2.1 Frame Disambiguation

In FrameNet, some target words can evoke multiple semantic frames. In order to extract the semantic structure of an ambiguous target word, the first step is to assign the correct frame to the target word in a given context. This task is similar with the word sense disambiguation task.

We select from the FN lexicon 556 target words that evoke at least two semantic frames and have at least five sentences annotated for each frame, and assemble a multi-class classifier for each ambiguous target word. As described in Figure 3, for this task we extract features used in word sense disambiguation (Florian et al., 2002), lexical features of the target word, and NAMED ENTITY FLAGS associated with the root node in a syntactic parse tree. For the rest of the ambiguous target words that have less than five sentences annotated we randomly choose a frame as being the correct frame in a given context.

2.2 Frame Element Identification

The idea of splitting the automatic semantic role labeling task into FE boundary detection and FE label classification was first proposed in (Gildea and Jurafsky, 2002) and then adopted by other works in this task. The problem of detecting the FE boundaries is cast as the problem of deciding whether or not a constituent is a valid candidate for a FE.

NO	TWFD	FEBD	GFLC	FELC	Feature Description
01	v				TW UNIGRAMS: The words, stem words and part of speech (POS) unigrams that are adjacent to target word expressions;
02	v				TW BIGRAMS: The words, stem words and POS bigrams that are adjacent to target word expressions;
03	v				TW WORD: The target word expression;
04			v	v	TW STEM: The stem word(s) of the target word expression;
05	v				TW POS: The POS of the target word;
06			v	v	TW CLASS: The lexical class of the target word, e.g. verb, noun, adjective;
07	v			v	NAMED ENTITY FLAGS: Set of binary features indicating whether a constituent contains, is contained or exactly identifies a named entity;
08	v				VERB WSD: If the target word is a verb, extract the head noun of the direct object and the prepositional object included in the verbal phrase;
09	v				NOUN WSD: If the target word is a noun, extract the head word of the verbal phrase that is in a verb−subject or verb−object relation with the noun;
10	v				ADJECTIVE WSD: If the target word is an adjective, extract the head noun that is modified by the adjective;
11			v	v	PHRASE TYPE: The syntactic category of the constituent;
12		v	v	v	DIRECTED PATH: Path in the syntactic parse tree between the constituent and the target word preserving the movement direction;
13				v	UNDIRECTED PATH: Same syntactic path as DIRECTED PATH without preserving the movement direction;
14		v			PARTIAL PATH: Path from the constituent to the earlier common ancestor of the target word and the constituent;
15		v	v	v	POSITION: Test whether the constituent contains the target word, or appears before or after the target word;
16			v	v	VOICE: Test if the verbal target word has active or passive construction;
17		v	v	v	HW: The head word of the constituent;
18			v	v	HW POS: The syntactic head POS of the constituent;
19				v	HW STEM: The stem word of the constituent's head word;
20				v	CW: The content word of the constituent computed as described in (Surdeanu et al., 2003);
21				v	CW POS: The POS corresponding to the content word;
22				v	CW STEM: Stemmed content word;
23			v	v	GOVERNING CATEGORY: Test whether the noun phrase constituents are dominated by verbal phrases or sentence phrases;
24		v			SYNTACTIC DISTANCE: The length of the syntactic path;
25				v	PP FIRST WORD: If the constituent is a prepositional phrase, return the first word in the phrase;
26				v	HUMAN: Test whether the constituent phrase is either a personal pronoun or a hyponym of first sense of PERSON synset in WordNet;
27				v	CONSTITUENTS NUMBER: The number of candidate FEs;
28				v	CONSTITUENTS LIST: Constituents labels list of the candidate FEs;
29				v	SAME CLAUSE: Test whether the constituent is in the same clause with the target word;
30				v	GF: The grammatical function of a candidate frame element;
31				v	GF LIST: The list of grammatical functions associated to the candidate FEs;
32		v	v	v	FRAME: The name of the semantic frame that is evoked by the target word;
33				v	NP SISTER: Determine whether the constituent has a noun phrase sister;
34				v	FIRST/LAST WORD: Return the first/last word of the constituent phrase;
35		v		v	FIRST/LAST POS: Return the first/last POS in the constituent;
36				v	LEFT/RIGHT SISTER LABEL: Return the left/right sibling constituent label;
37				v	LEFT/RIGHT SISTER HEAD: Return the left/right sibling head word;
38				v	LEFT/RIGHT SISTER STEM HEAD: Return the left/right sibling stemmed head word;
39				v	LEFT/RIGHT SISTER POS HEAD: Return the left/right sibling head POS;
40				v	TW STEM & HW STEM: Join of TW STEM and HW STEM;
41				v	TW STEM & PHRASE TYPE: Join of TW STEM and PHRASE TYPE;
42				v	VOICE & POSITION: Join of VOICE and POSITION.

Figure 3: Feature set for extracting frame semantic structures.

We consider a binary classifier over the entire FN data and extract features for each constituent from a syntactic parse tree. Because this experimental setup allows training the binary classifier on a large set of examples, the best feature combination consists of a restrained number of features. Most of these features are from the set proposed by (Gildea and Jurafsky, 2002). Another feature that improved the prediction of FE boundaries in every feature selection experiment is the FRAME feature. Since the frame disambiguation is executed before the FE boundary detection in the pipeline architecture, we can use the FRAME feature at this step. This feature helps the binary classifier distinguish between frame element structures from different semantic frames.

2.3 Grammatical Function Classification

Once we identify the candidate boundaries for frame elements, the next step is to assign the grammatical functions to these boundaries. In FrameNet, the grammatical functions represent the manner in which the frame elements satisfy grammatical constraints with respect to the target word.

For this task we train a multi-class classifier over the entire lexicon to predict seven categories of GFs that exist in FN. In addition, we assign the NULL category for those FEs that double as target words. The features are extracted only for the constituents that are identified as FEs in the previous FE boundary identification sub-task. The best feature set in this phase includes the features proposed by (Gildea and Jurafsky, 2002) and the FRAME feature.

2.4 Frame Element Classification

The task of FE classification is to assign FE labels to every constituent identified as FE. In order to predict the frame elements, which are locally defined for each semantic frame, we built 489 multi-class classifiers, where each classifier corresponds to a frame in FrameNet. This partitioning of the FN lexicon has the advantage of increasing the overall classification performance and efficiently learning the frame elements labels. On the other hand, this approach suffers from the lack of annotated data in some frames and hence it requires using a large set of features.

The advantage of designing the classifiers in a pipeline architecture is best illustrated in this subtask. Some of the most effective features for FE classification are extracted using information from previous sub-tasks: FRAME feature is made available by the TWFD sub-task, CONSTITUENTS NUMBER and CONSTITUENTS LIST are made available by the FEBD sub-task, and GF and GF LIST are made available by the GFLC sub-task.

3 Experimental Results

We report experimental results on all four classification sub-tasks. In our experiments we trained two types of classification models for each sub-task: SVM and ME. In order to optimize the performance measure of each sub-task and to find the best configuration of classification models we used 20% of the sub-tasks training data as validation data. Table 1 lists the best configuration of classification models as well as the best sub-task results when running the experiments on the validation data. For frame disambiguation, we obtained 76.71% accuracy compared to a baseline of 60.72% accuracy that always predicts the most annotated frame for each of the 556 target words. The results for GFLC and FELC sub-tasks listed in Table 1 were achieved by using gold FE boundaries.

Task	*Best Model*	*Accuracy*		
Frame Disambiguation	SVM	76.71		
GF Label Classification	ME	96.00		
FE Label Classification	ME	88.93		
		Precision	*Recall*	*F1−measure*
FE Boundary Detection	SVM	73.65	87.08	79.80

Table 1: Task results on the validation set.

The SemEval–2007 organizers provided fully annotated training files, a scorer to evaluate these training files, and testing files containing flat sentences. In the evaluation process, a semantic dependency graph corresponding to a fully system annotated sentence is created and then matched with its gold dependency graph. The matching process not only evaluates every semantic structure of a target word, but also considers frame-to-frame and FE-to-FE graph relations between the semantic structures. In addition, various scoring options were considered: exact or partial frame matching, partial credit for evaluating the named entities, evaluation of the flat frame elements labels, and an option for matching only the frames in evaluation. The evaluation for flat frame elements labels is similar with the evaluation performed at Senseval-3. The only difference is that for this scorer the FE boundaries must match exactly.

In Table 2, we present the averaged precision, recall and F1 measures for evaluating the semantic dependency graphs and detecting the semantic frames on the testing files. The *"Options"* column represents the configuration parameters of the scorer: (E)xact/(P)artial frame matching, semantic (D)ependency or (L)abels only evaluation, and (Y)es/(N)o named entity evaluation.

Options	*Semantic Dependency Evaluation*			*Frame Detection Evaluation*		
	Precision	*Recall*	*F1−measure*	*Precision*	*Recall*	*F1−measure*
E L Y	51.10	27.74	35.88	69.16	42.73	52.71
P L Y	55.56	30.19	39.04	77.82	48.09	59.32
E D Y	50.29	27.05	35.11	71.69	44.43	54.74
P D Y	54.78	29.48	38.26	80.35	49.79	61.35
E L N	51.85	27.59	35.94	69.16	42.73	52.71
P L N	56.59	30.14	39.25	77.82	48.09	59.32
E D N	51.38	26.95	35.29	71.69	44.43	54.74
P D N	56.13	29.45	38.57	80.35	49.79	61.35

Table 2: System results on the test set.

Although the system achieved good precision scores on the test data, the recall values caused the system to obtain unsatisfactory F1-measure values. We expect that the recall will increase by considering various heuristics for a better mapping of the frame elements to constituents in parse trees.

4 Conclusions

We described a system that participated in SemEval–2007 for the task of extracting frame semantic structures. We showed that a pipeline architecture of the SVM and ME classifiers as well as an adequate selection of the classification models can improve the performance measures of each sub-task.

References

Collin F. Baker, Charles J. Fillmore, and John B. Lowe. 1998. The Berkeley FrameNet project. In *Proceedings of the COLING-ACL*, Montreal, Canada.

Cosmin Adrian Bejan, Alessandro Moschitti, Paul Morărescu, Gabriel Nicolae, and Sanda Harabagiu. 2004. Semantic Parsing Based on FrameNet. In *Senseval-3: Workshop on the Evaluation of Systems for the Semantic Analysis of Text.*

Radu Florian, Silviu Cucerzan, Charles Schafer, and David Yarowsky. 2002. Combining classifiers for word sense disambiguation. *Natural Language Engineering*.

Daniel Gildea and Daniel Jurafsky. 2002. Automatic Labeling of Semantic Roles. *Computational Linguistic*.

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

Mihai Surdeanu, Sanda M. Harabagiu, John Williams, and Paul Aarseth. 2003. Using predicate-argument structures for information extraction. In *Proceedings of ACL*.

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