SENSEVAL-3: Third International Workshop on the Evaluation of Systems
for the Semantic Analysis of Text, Barcelona, Spain, July 2004
Association for Computational Linguistics

SENSEVAL Automatic Labeling of Semantic Roles using Maximum Entropy Models

Namhee Kwon
Information Science Institute
University of Southern California
4676 Admiralty Way
Marina del Rey, CA 90292
nkwon@isi.edu

Michael Fleischman
Messachusetts Institute of Technology
77 Massachusetts Ave
Cambridge, MA 02139
mbf@mit.edu

Eduard Hovy
Information Science Institute
University of Southern California
4676 Admiralty Way
Marina del Rey, CA 90292
hovy@isi.edu

Abstract

As a task in SensEval-3, Automatic Labeling of Semantic Roles is to identify frame elements within a sentence and tag them with appropriate semantic roles given a sentence, a target word and its frame. We apply Maximum Entropy classification with feature sets of syntactic patterns from parse trees and officially attain 80.2% precision and 65.4% recall. When the frame element boundaries are given, the system performs 86.7% precision and 85.8% recall.

1 Introduction

The Automatic Labeling of Semantic Roles track in SensEval-3 focuses on identifying frame elements in sentences and tagging them with their appropriate semantic roles based on FrameNet[1].

For this task, we extend our previous work (Fleischman et el., 2003) by adding a sentence segmentation step and by adopting a few additional feature vectors for Maximum Entropy Model. Following the task definition, we assume the frame and the lexical unit of target word are known although we have assumed only the target word is known in the previous work.

2 Model

We separate the problem of FrameNet tagging into three subsequent processes: 1) sentence segmentation 2) frame element identification, and 3) semantic role tagging. We assume the frame element (FE) boundaries match the parse constituents, so we segment a sentence based on parse constituents. We consider steps 2) and 3) as classification problems. In frame element identification, we use a binary classifier to determine if each parse constituent is a FE or not, while, in semantic role tagging, we classify each identified FE into its appropriate semantic role.[2] Figure 1 shows the sequence of steps.

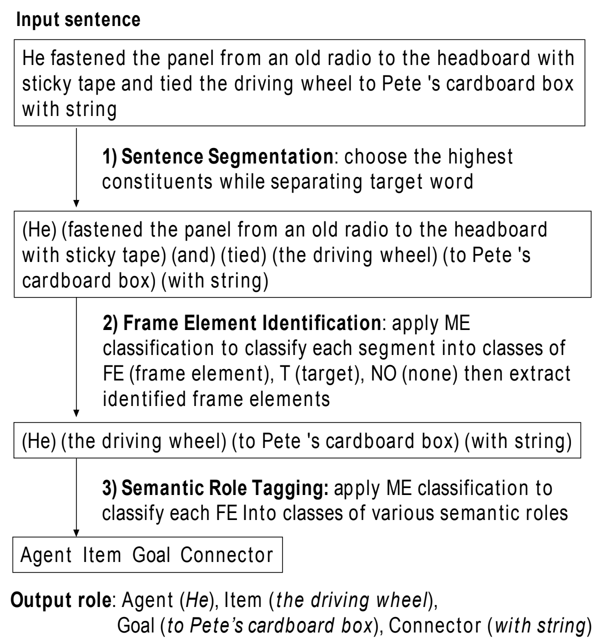

Fig. 1. The sequence of steps on a sample sentence having a target word "tied".

We train the ME models using the GIS algorithm (Darroch and Ratcliff, 1972) as implemented in the YASMET ME package (Och, 2002). We use the YASMET MEtagger (Bender et al. 2003) to perform the Viterbi search for choosing the most probable tag sequence for a sentence using the probabilities computed during training. Feature weights are smoothed using Gaussian priors with mean 0 (Chen and Rosenfeld, 1999).

2.1 Sentence Segmentation

We segment a sentence into a sequence of non-overlapping constituents instead of all individual constituents. There are a number of advantages to applying sentence segmentation before FE

[1] http://www.icsi.berkeley.edu/~framenet

[2] We are currently ignoring null instantiations.

boundary identification. First, it allows us to utilize sentence-wide features for FE identification. The sentence-wide features, containing dependent information between frame element such as the previously identified class or the syntactic pattern, have previously been shown to be powerful features for role classification (Fleischman et al., 2003). Further, it allows us to reduce the number of candidate constituents for FE, which reduces the convergence time in training.

The constituents are derived from a syntactic parse tree[3]. Although we need to consider all combinations of various level constituents in a parse tree, we know the given target word should be a separate segment because a target word is not a part of other FEs.[4] Since most frame elements tend to be in higher levels of the parse tree, we decide to use the highest constituents (the parse constituents having the maximum number of words) while separating the target word. Figure 2 shows an example of the segmentation for an actual sentence in FrameNet with the target word "tied".

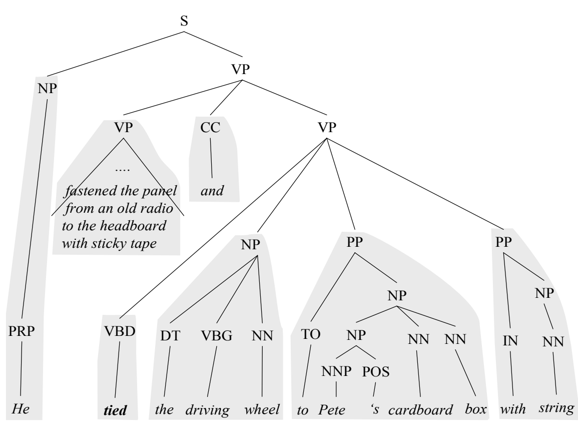

Fig. 2. A sample sentence segmentation: "tied" is the target predicate, and the shaded constituent represents each segment.

However, this segmentation reduces the FE coverage of constituents (the number of constituents matching frame elements). In Table 1, "individual constituents" means a list of all constituents, and "Sentence segmentation" means a sequence of non-overlapping constituents that are taken in our work. We can regard 85.8% as the accuracy of the parser.

[3] We use Charniak parser : http://www.cs.brown.edu/people/ec/#software

[4] Although 17% of constituents are both a target and a frame element, there is no case that a target is a part of a frame element.

Method	Number of constituents	FE coverage (%)
Individual constituents	342,245	85.8
Sentence segmentation	66,401	79.5

Table 1. FE coverage for the test set.

2.2 Frame Element Identification

Frame element identification is executed for segments to classify into the classes on FE, Target, or None. When a constituent is both a target and a frame element, we set it as a frame element when training because we are interested in identifying frame elements not a target.

The initial features are adopted from (Gildea and Juraksky 2002) and (Fleischman, Kwon, and Hovy 2003), and a few additional features are also used. The features are:

- **Target predicate (target):** The target is the principal lexical item in a sentence.
- **Target lexical name (lexunit):** The formal lexical name of target predicate is the string of the original form of target word and grammatical type. For example, when the target is "tied", the lexical name is "tie.v".
- **Target type (ltype):** The target type is a part of lexunit representing verb, noun, or adjective. (e.g. "v" for a lexunit "tie.v")
- **Frame name (frame):** The semantic frame is defined in FrameNet with corresponding target.
- **Constituent path (path):** From the syntactic parse tree of a sentence, we extract the path from each constituent to the target predicate. The path is represented by the nodes through which one passes while traveling up the tree from the constituent and then down through the governing category to the target word. For example, "the driving wheel" in the sentence of Figure 2 has the path, NP ↑ VP ↓ VBD.
- **Partial path (ppath):** The partial path is a variation of path, and it produces the same path as above if the constituent is under the same "S" as target word, if not, it gives "nopath".
- **Syntactic Head (head):** The syntactic head of each constituent is obtained based on Michael Collins's heuristic method[5]. When the head is a proper noun, "proper-noun" substitutes for the real head. The decision as to whether the head is a proper noun is made based on the part of speech tags used in the parse tree.

[5] http://www.ai.mit.edu/people/mcollins/papers/heads

- **Phrase Type (pt):** The syntactic phrase type (e.g., NP, PP) of each constituent is also extracted from the parse tree. It is not the same as the manually defined PT in FrameNet.
- **Logical Function (lf):** The logical functions of constituents in a sentence are simplified into three values: *external argument*, *object argument*, *other*. When the constituent's phrase type is NP, we follow the links in the parse tree from the constituent to the ancestors until we meet either S or VP. If the S is found first, we assign *external argument* to the constituent, and if the VP is found, we assign *object argument.* Otherwise, *other* is assigned.
- **Position (pos):** The position indicates whether a constituent appears before or after the target predicate.
- **Voice (voice):** The voice of a sentence (active, passive) is determined by a simple regular expression over the surface form of the sentence.
- **Previous class (c_n):** The class information of the n^{th}-previous constituent (Target, FE, or None) is used to exploit the dependency between constituents. During training, this information is provided by simply looking at the true class of the constituent occurring *n*-positions before the target element. During testing, the hypothesized classes are used for Viterbi search.

Feature Set	Example Functions
f(c, lexunit)	*f(c, tie.v) = 1*
f(c, pt, pos, voice)	*f(c, NP,after,active) = 1*
f(c, pt, lf)	*f(c, ADVP,obj) = 1*
f(c, pt_-1, lf_-1)	*f(c, VBD_-1, other_-1) = 1*
f(c, pt_1, lf_1)	*f(c, PP_1, other_1) = 1*
f(c, head)	*f(c, wheel) = 1*
f(c, head, frame)	*f(c, wheel, Attaching) = 1*
f(c, path)	*f(c, NP ↑ VP ↓ VBD) = 1*
f(c, path_-1)	*f(c, VBD_-1) = 1*
f(c, path_1)	*f(c, PP ↑ VP ↓ VBD_1) = 1*
f(c, target)	*f(c, tied) = 1*
f(c, ppath)	*f(c, NP ↑ VP ↓ VBD) = 1*
f(c, ppath, pos)	*f(c,NP ↑ VP ↓ VBD, after) = 1*
f(c, ppath_-1, pos_-1)	*f(c, VBD_-1,after) = 1*
f(c ,ltype, ppath)	*f(c, v, NP ↑ VP ↓ VBD) = 1*
f(c ,ltype, path)	*f(c, v, NP ↑ VP ↓ VBD) = 1*
f(c ,ltype, path_-1)	*f(c, v,VBD_-1) = 1*
f(c frame)	*f(c, Attaching) = 1*
f(c, frame, c_-1)	*f(c, Attaching, T_-1) = 1*
f(c,frame, c_-2,c_-1)	*f(c, Attaching,NO_-2,T_-1)=1*

Table 2. Feature sets used in ME frame element identification. Example functions of "the driving wheel" from the sample sentence in Fig.2.

The combinations of these features that are used in the ME model are shown in Table 2. These feature sets contain the previous or next constituent's features, for example, *pt_-1* represents the previous constituent's phrase type and *lf_1* represents the next constituent's logical function.

2.3 Semantic Role Classification

Semantic role classification is executed only for the constituents that are classified into FEs in the previous FE identification phase by employing Maximum Entropy classification.

In addition to the features in Section 2.2, two more features are applied.

- **Order (order):** The relative position of a frame element in a sentence is given. For example, the sentence from Figure 2 has four frame elements, where the element "He" has order 0, while "with string" has order 3.
- **Syntactic pattern (pat):** The sentence level syntactic pattern is generated from the parse tree by considering the phrase type and logical functions of each frame element in the sentence. In the example sentence in Figure 2, "He" is an external argument Noun Phrase, "tied" is a target predicate, and "the driving wheel" is an object argument Noun Phrase. Thus, the syntactic pattern associated with the sentence is [NP-ext, target, NP-obj, PP-other, PP-other].

Table 3 shows the list of feature sets used for the ME role classification.

Feature Set	
f(r, lexunit)	*f(r, pt, lf)*
f(r, target)	*f(r, pt_-1, lf_-1)*
f(r, pt, pos, voice)	*f(r, pt_1, lf_1)*
f(r, head)	*f(r, order, syn)*
f(r, head, lexunit)	*f(r, lexunit, order, syn)*
f(r, head, frame)	*f(r, pt, pos, voice, lexunit)*
f(r, frame, r_-1)	*f(r, frame, r_-2,r_-1)*
f(r, frame,r_-3, r_-2,r_-1)	

Table 3. Feature sets used in role classification.

3 Results

SensEval-3 provides the following data set: training set (24,558 sentences/ 51,323 frame elements/ 40 frames), and test set (8,002 sentences/ 16,279 frame elements/ 40 frames). We submit two sets to SensEval-3, one (test A) is the output of all above processes (identifying frame elements and tagging them given a sentence), and the other (test B) is to tag semantic roles given frame elements.

For test B, we attempt the role classification for all frame elements including frame elements not matching the parse tree constituents. Although there are frame elements that have two different semantic roles, we ignore those cases and assign one semantic role per frame element. This explains why test B shows 99% attempted frame elements. The attempted number for test A is the number of frame elements identified by our system. Table 4 shows the official scores for these tests.

Test	Prec.	Overlap	Recall	Attempted
Test A	80.2	78.4	65.4	81.5
Test B	86.7	86.6	85.8	99.0

Table 4. Final SensEval-3 scores for the test set.

In the official evaluation, the precision and recall are calculated by counting correct roles that overlap even in only one word with the reference set. Overlap score shows how much of an actual FE is identified as an FE not penalizing wrongly identified part. Since this evaluation is so lenient, we perform another evaluation to check exact matches.

Method	FE boundary Identification		FE boundary Identification & Role labeling	
	Prec	Rec	Prec	Rec
Test A	80.3	66.1	71.1	58.5
Test B	100.0	99.0	86.7	85.8

Table 5. Exact match scores for the test set.

4 Discussion and Conclusion

Due to time limit, we've not done many experiments with different feature sets or thresholds in ME classification. We expect that recall will increase with lower thresholds especially in lenient evaluation and the final performance will increase by optimizing parameters.

References

O. Bender, K. Macherey, F.J. Och, and H. Ney. 2003. Comparison of Alignment Templates and Maximum Entropy Models for Natural Language Processing. Proc. of *EACL-2003*. Budapest, Hungary.

A. Berger, S. Della Pietra and V. Della Pietra, 1996. A Maximum Entropy Approach to Natural Language Proc. of *Computational Linguistics,* vol. 22, no. 1.

E. Charniak. 2000. A Maximum-Entropy-Inspired Parser. Proc. of *NAACL-2000*, Seattle, USA.

S.F. Chen and R. Rosenfeld. 1999. A Gaussian Prior for Smoothing Maximum Entropy Models. *Technical Report CMUCS-99-108*, Carnegie Mellon University.

J. N. Darroch and D. Ratcliff. 1972. *Generalized Iterative Scaling for Log-Linear Models*. Annals of Mathematical Statistics, 43:1470-1480.

C.Fillmore 1976. Frame Semantics and the Nature of Language. *Annals of the New York Academy of Science Conference on the Origin and Development of Language and Speech,* Volume 280 (pp. 20-32).

M. Fleischman, N. Kwon, and E. Hovy. 2003. Maximum Entropy Models for FrameNet Classification. Proc. of *Empirical Methods in Natural Language Processing conference (EMNLP)*, 2003. Sapporo, Japan.

D. Gildea and D. Jurafsky. 2002. Automatic Labeling of Semantic Roles. *Computational Linguistics*, 28(3) 245-288 14.

F.J. Och. 2002. Yet Another Maxent Toolkit: YASMET www-i6.informatik.rwth-aachen.de/Colleagues/och/.

C. Thompson, R. Levy, and C. Manning. 2003. A Generative Model for FrameNet Semantic Role Labeling. Proc. of *the Fourteenth European Conference on Machine Learning*, 2003. Croatia.