# Open Text Semantic Parsing Using FrameNet and WordNet

**Lei Shi and Rada Mihalcea**
Department of Computer Science and Engineering
University of North Texas
leishi@unt.edu, rada@cs.unt.edu

## Abstract

This paper describes a rule-based semantic parser that relies on a frame dataset (FrameNet), and a semantic network (WordNet), to identify semantic relations between words in open text, as well as shallow semantic features associated with concepts in the text. Parsing semantic structures allows semantic units and constituents to be accessed and processed in a more meaningful way than syntactic parsing, moving the automation of understanding natural language text to a higher level.

## 1 Introduction

The goal of the semantic parser is to analyze the semantic structure of a natural language sentence. Similar in spirit with the syntactic parser – whose goal is to parse a valid natural language sentence into a parse tree indicating how the sentence can be syntactically decomposed into smaller syntactic constituents – the purpose of the semantic parser is to analyze the structure of sentence meaning. Sentence meaning is composed by *entities* and *interactions* between entities, where entities are assigned semantic *roles*, and can be further modified by other *modifiers*. The meaning of a sentence is decomposed into smaller semantic units connected by various semantic relations by the principle of compositionality, and the parser represents the semantic structure – including semantic units as well as semantic relations, connecting them into a formal format.

One major problem faced by many natural language understanding applications that rely on syntactic analysis of text, is the fact that similar syntactic patterns may introduce different semantic interpretations. Likewise, similar meanings can be syntactically realized in many different ways. The semantic parser attempts to solve this problem, and produces a syntax-independent representation of sentence meaning, so that semantic constituents can be accessed and processed in a more meaningful and flexible way, avoiding the sometimes rigid interpretations produced by a syntactic analyzer. For instance, the sentences *I boil water* and *water boils* contain a similar relation between *water* and *boil*, even though they have different syntactic structures.

In this paper, we describe the main components of the semantic parser, and illustrate the basic procedures involved in parsing semantically open text. Our semantic parser departs from current approaches in statistics-based annotations of semantic structures. Instead, we are using publicly available lexical resources (FrameNet and WordNet) as a starting point to derive rules for a rule-based semantic parser.

## 2 Semantic Structure

Semantics is the denotation of a string of symbols, either a sentence or a word. Similar to a syntactic parser, which shows how a larger string is formed by smaller strings from a formal point of view, the semantic parser shows how the denotation of a larger string – sentence, is formed by denotations of smaller strings – words. Syntactic relations can be described using a set of rules about how a sentence string is formally generated using word strings. Instead, semantic relations between semantic constituents depend on our understanding of the world, which is across languages and syntax.

We can model the sentence semantics as describing entities and interactions between entities. Entities can represent physical objects, as well as time, places, or ideas, and are usually formally realized as nouns or noun phrases. Interactions, usually realized as verbs, describe relationships or interactions between participating entities. Note that a participant can also be an interaction, which can be regarded as an entity nominalized from an interaction. We assign semantic roles to participants and their semantic relations are identified by the case frame introduced by their interaction. In a sentence, participants and interactions can be further modified by various modifiers, including descriptive modifiers that describe attributes such as *drive slowly*, restrictive modifiers that enforce a general denotation to become more specific such as *musical instrument*, referential modifiers that indicate particular instances such as *the pizza I ordered*. Other semantic relations can also be identified, such as coreference, complement, and others. Based on the principle of compositionality, the sentence semantic structure is recursive, similar to a tree.

Note that the semantic parser analyzes shallow-level semantics, which is derived directly from linguistic knowledge, such as rules about semantic role assignment, lexical semantic knowledge, and syntactic-semantic mappings, without taking into account any context or common sense knowledge. Hence, the parser can be used as an intermediate semantic processing level before higher levels of text understanding.

## 3 Knowledge Bases for Semantic Parsing

The parser relies on two main types of knowledge – about words, and about relations between words. The first type of knowledge is drawn from WordNet – a large lexical database with rich information about words and concepts. We refer to this as *word-level knowledge*. The latter is derived from FrameNet – a resource that contains information about different situations, called *frames*, in which semantic relations are syntactically realized in natural language sentences. We call this *sentence-level knowledge*. In addition to these two lexical knowledge bases, the parser also utilizes a set of manually defined rules, which encode mappings from syntactic structures to semantic relations, and which are used to handle those structures not explicitly addressed by FrameNet or WordNet. In this section, we describe the type of information extracted from these knowledge bases, and show how this information is encoded in a format accessible to the semantic parser.

### 3.1 Sentence Level Knowledge

FrameNet (Johnson et al., 2002) provides the knowledge needed to identify case frames and semantic roles. FrameNet is based on the theory of frame semantics, and defines a sentence level ontology. In frame semantics, a *frame* corresponds to an *interaction* and its *participants*, both of which denote a scenario, in which participants play some kind of roles. A frame has a name, and we use this name to identify the semantic relation that groups together the semantic roles. Nouns, verbs and adjectives can be used to identify frames.

Each annotated sentence in FrameNet exemplifies a possible syntactic realization for the semantic roles associated with a frame for a given target word. By extracting the syntactic features and corresponding semantic roles from all annotated sentences in the FrameNet corpus, we are able to automatically build a large set of rules that encode the possible syntactic realizations of semantic frames.

#### 3.1.1 Rules Learned from FrameNet

FrameNet data "is meant to be lexicographically relevant, not statistically representative" (Johnson et al., 2002), and therefore we are using FrameNet as a starting point to derive rules for a rule-based semantic parser.

To build the rules, we are extracting several syntactic features. Some are explicitly encoded in FrameNet, such as the *grammatical function* (GF) and *phrase type* (PT) features. In addition, other syntactic features are extracted from the sentence context. One such feature is the *relative position* (RP) to the target word. Another feature is the *voice* of the sentence. If the phrase type is prepositional phrase (PP), we also record the actual *preposition* that precedes the phrase.

After we extract all these syntactic features, the semantic role is appended to the rule, which creates a mapping from syntactic features to semantic roles.

Feature sets are arranged in a list, the order of which is identical to that in the sentence. Altogether, the rule for a possible realization of a frame exemplified by a tagged sentence is an ordered sequence of syntactic features with their semantic roles. For example, the corresponding formalized rule for the sentence *I had chased Selden over the moor* is:

```
[active, [ext,np,before,theme], [obj,np,after,goal],
         [comp,pp,after,over,path]]
```

In FrameNet, there are multiple annotated sentences for each frame to demonstrate multiple possible syntactic realizations. All possible realizations of a frame are collected and stored in a list for that frame, which also includes the target word, its syntactic category, and the name of the frame. All the frames defined in FrameNet are transformed into this format, so that they can be easily handled by the rule-based semantic parser.

### 3.2 Word Level Knowledge

WordNet (Miller, 1995) is the resource used to identify shallow semantic features that can be attached to lexical units. For instance, attribute relations, adjective/adverb classifications, and others, are semantic features extracted from WordNet and stored together with the words, so that they can be directly used in the parsing process.

All words are uniformly defined, regardless of their class. Features are assigned to each word, including syntactic and shallow semantic features, indicating the functions played by the word. Syntactic features are used by the feature-augmented syntactic analyzer to identify grammatical errors and produce syntactic information for semantic role assignment. Semantic features encode lexical semantic information extracted from WordNet that is used to determine semantic relations between words in various situations.

Features can be arbitrarily defined, as long as there are rules to handle them. The features we define encode information about the syntactic category of a word, number and countability for nouns, transitivity and form for verbs, type, degree, and attribute for adjectives and adverbs, and others.

For example, for the adjective *slow*, the entry in the lexicon is defined as:

```
lex(slow,W):- W= [parse:slow, cat:adj, attr:speed,
                  degree:base, type:descriptive].
```

Here, the category (*cat*) is defined as adjective, the *type* is descriptive, *degree* is base form. We also record the *attr* feature, which is derived from the *attribute* relation in WordNet, and links a descriptive adjective to the attribute (noun) it modifies, such as *slow* → *speed*.

## 4 The Semantic Parser

The parsing algorithm is implemented as a rule-based system. The general procedure of semantic parsing consists of three main steps: (1) syntactic parsing into an intermediate format, using a feature-augmented syntactic parser, and assignment of shallow semantic features; (2) semantic role assignment; (3) application of default rules.

### 4.1 Feature Augmented Syntactic/Semantic Analyzer

The semantic parser is based on dependencies between words that are identified using a structure analyzer. The analyzer generates an intermediate format, where target words and syntactic arguments are explicitly identified, so that they can be matched against the rules derived from FrameNet. The intermediate format also encodes some shallow semantic features, including word level semantics (e.g. attribute, gender), and semantic relations that have direct syntactic correspondence (e.g. modifier types). The function of the sentence is also identified, as *assertion*, *query*, *yn-query*, *command*.

The analyzer is based on a feature augmented grammar, and has the capability of detecting if a sentence is grammatically correct (unlike statistical parsers, which attempt to parse any sentence, regardless of their well-formness). Constituents are assigned with features, and the grammar consists of a set of rules defining how constituents can connect to each other, based on the values of their features.

Since features can contain both syntactic and semantic information, the analyzer can reject some grammatically incorrect sentences such as: *I have much apples*, *You has my car*, or even some semantically incorrect sentences: *The technology is very military*[1].

### 4.2 Semantic Role Assignment

In the process of semantic role assignment, we first start by identifying all possible frames, according to the *target* word. Next, a matching algorithm is used to find the most likely match among all rules derived for these frames, to identify the correct frame (if several are possible), and assign semantic roles.

In a sentence describing an interaction, we usually select the verb or predicative adjective as the *target* word, which triggers the sentence level frame. A noun can also play the role of target word, but only within the scope of the noun phrase it belongs to, and it can be used to assign semantic roles only to its modifiers.

The matching algorithm relies on a scoring function to evaluate the similarity between two sequences of syntactic features. The matching starts from left to right. Whenever an exact match is found, the score will be increased by 1. It should be noted that the search sequence is uni-directional which means that once you find a match, you can go ahead to check features to the right, but you cannot go back to check rules you have already checked. This guarantees that syntactic features are matched in the right order, and the order of sequence in the rule is maintained. Since the frame of a target word may have multiple possible syntactic realizations, which are exemplified by different sentences in the corpus, we try to match the syntactic features in the intermediate format with all the rules available for the target word, and compare their matching scores. The rule with the highest score is selected, and used for semantic role assignment. Through this scoring scheme, the matching algorithm tries to maximize the number of syntactic realizations for semantic roles defined in FrameNet rules.

Notice that the semantic role assignment is performed recursively, until all roles within frames triggered by all target words are assigned.

#### 4.2.1 Walk-Through Example

Assume the following two rules, derived from FrameNet for the target word *come*:

```
1:[[ext,np,before,active,theme],
   [obj,np,after,active,goal],
   [comp,pp,after,active,by,mode_of_transportation]]

2:[[ext,np,before,active,theme],
   [obj,np,after,active,goal],
   [comp,pp,after,active,from,source]]
```

And the sentences:

```
A: I come here by train.

B: I come here from home.
```

The syntactic features identified by the syntactic analyzer for these two sentences are:

```
A':[[ext,np,before,active], [obj,np,after,active],
$[$comp,pp,after,active,by]]

B':[[ext,np,before,active], [obj,np,after,active],
$[$comp,pp,after,active,from]]
```

Using the matching/scoring algorithm, the score for matching A' to rule 1 is determined as 3, and to rule 2 as 2. Hence, the matching algorithm selects rule 1, and the semantic role for *train* is *mode_of_transportation*. Similarly, when we match B' to rule 1, we obtain a score of 2, and a larger score of 3 for matching with rule 2. Therefore, for the second case, the role assigned to *home* is *source*.

### 4.3 Applying Default Rules

In a sentence, semantic roles are played by the subject, objects, and the prepositional phrases attached to the interaction described by the sentence. However, FrameNet defines roles only for some of these elements, and therefore the meaning of some sentence constituents cannot be determined using the rules extracted from FrameNet. In order to handle these constituents, and allow for a complete semantic interpretation of the sentence, we have defined a set of default rules that are applied as a last step in the process of semantic parsing. For example, FrameNet defines a role for the prepositional phrase *on him* in "I depend on him", but it does not define a role for the phrase *on the street* in "I walk on the street". To handle the interpretation of this phrase,

[1] Since *military* is not a descriptive adjective, it cannot be connected to the degree modifier *very*.

we apply the default rule that “on something” modifies the *location* attribute of an interaction.

We have defined about 100 such default rules, which are assigned in the last step of the semantic parsing process, if no other rule could be applied in previous steps. After this step, the semantic structure of the sentence is produced.

## 5 Parser Output and Evaluation

The semantic parser is demonstrated in this conference, which is perhaps the best evaluation we can offer. We illustrate here the output of the semantic parser on a natural language sentence, and show the corresponding semantic structure and tree. For example, for the sentence *I like to eat Mexican food because it is spicy*, the semantic parser produces the following encoding of sentence type, frames, semantic constituents and roles, and various attributes and modifiers:

```
T = assertion
P =
[[experiencer, [[entity, [i], reference(first)],
      [modification(attribute), quantity(single)]]],
  [interaction(experiencer\_subj),[love]],
      [modification(attribute), time(present)],
  [content, [
       [interaction(ingestion), [eat]],
       [ingestibles, [entity, [food]]
            [[modification(restriction), [mexican]],
  ]]]],
  [reason, [[agent, [[entity, [it], reference(third)],
       [modification(attribute), quantity(single)]]],
       [description,
           [modification(attribute), time(present)]],
       [modification(attribute), taste\_property(spicy)]]]]
]
```

The corresponding semantic tree is shown in Figure 1.

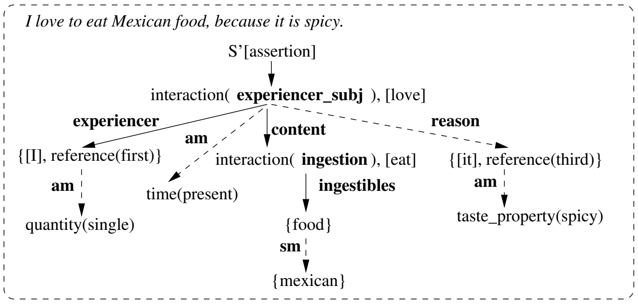


Figure 1: Semantic parse tree (*am* = attributive modifier, *rm* = referential modifier, *sm* = restrictive modifier)

We have conducted evaluations of the semantic role assignment algorithm on 350 sentences randomly selected from FrameNet. The test sentences were removed from the FrameNet corpus, and the rules-learning procedure described earlier in the paper was invoked on this reduced corpus. All test sentences were then semantically parsed, and full semantic annotations were produced for each sentence. Notice that the evaluation is conducted only for semantic role assignment – since this is the only information available in FrameNet. The other semantic annotations produced by the parser (e.g. attribute, gender, countability) are not evaluated at this point, since there are no hand-validated annotations of this kind available in current resources.

Both frames and frame elements are automatically identified by the parser. Out of all the elements correctly identified, we found that 74.5% were assigned with the correct role (this is therefore the *accuracy* of role assignment), which compares favorably with previous results reported in the literature for this task. Notice also that since this is a rule-based approach, the parser does not need large amounts of annotated data, but it works well the same for words for which only one or two sentences are annotated.

## 6 Related Work

All previous work in semantic parsing has exclusively focused on labeling semantic roles, rather than analyzing the full structure of sentence semantics, and is usually based on statistical models - e.g. (Gildea and Jurafsky, 2000), (Fleischman et al., 2003). To our knowledge, there was no previous attempt on performing semantic annotations using alternative rule-based algorithms. However, a rule-based approach is closer to the way humans interpret the semantic structure of a sentence. Moreover, as mentioned earlier, the FrameNet data is not meant to be “statistically representative”, but rather illustrative for various language constructs, and therefore a rule-based approach is more suitable for this lexical resource.

## 7 Conclusions

We described a rule-based approach to open text semantic parsing. The semantic parser has the capability to analyze the semantic structure of a sentence, and show how the meaning of the entire sentence is composed of smaller semantic units, linked by various semantic relations. The parsing process relies on rules derived from a frame dataset (FrameNet) and a semantic network (WordNet). We believe that the semantic parser will prove useful for a range of language processing applications that require knowledge of text meaning, including word sense disambiguation, information extraction, question answering, machine translation, and others.

## References

M. Fleischman, N. Kwon, and E. Hovy. 2003. Maximum entropy models for FrameNet classification. In *Proceedings of 2003 Conference on Empirical Methods in Natural Language Processing EMNLP-2003*, Sapporo, Japan.

D. Gildea and D. Jurafsky. 2000. Automatic labeling of semantic roles. In *Proceedings of the 38th Annual Conference of the Association for Computational Linguistics (ACL-00)*, pages 512–520, Hong Kong, October.

C. Johnson, C. Fillmore, M. Petruck, C. Baker, M. Ellsworth, J. Ruppenhofer, and E. Wood. 2002. *FrameNet: Theory and Practice*. http://www.icsi.berkeley.edu/ framenet.

G. Miller. 1995. Wordnet: A lexical database. *Communication of the ACM*, 38(11):39–41.