

# Using Frame Semantics in Natural Language Processing

**Apoorv Agarwal**

Dept. of Computer Science  
Columbia University  
New York, NY

apoorv@cs.columbia.edu

**Daniel Bauer**

Dept. of Computer Science  
Columbia University  
New York, NY

bauer@cs.columbia.edu

**Owen Rambow**

CCLS  
Columbia University  
New York, NY

rambow@ccls.columbia.edu

## Abstract

We summarize our experience using FrameNet in two rather different projects in natural language processing (NLP). We conclude that NLP can benefit from FrameNet in different ways, but we sketch some problems that need to be overcome.

## 1 Introduction

We present two projects at Columbia in which we use FrameNet. In these projects, we do not develop basic NLP tools for FrameNet, and we do not develop FramNets for new languages: we simply use FrameNet or a FrameNet parser in an NLP application. The first application concerns the extraction of social networks from narrative texts. The second application aims at generating three-dimensional pictures from textual descriptions.

The applications are very different: they differ in terms of their goals, and they differ in terms of how they use FrameNet. However, they have in common that they can use FrameNet because it provides a particular level of semantic abstraction which is suited for both applications. Consider verbs of saying, such as *declare*, *deny*, *mention*, *remark*, *tell*, or *say*: they do not have the same meaning. However, they share enough common meaning, and in particular they share the same set of participants, so that for our two applications they can be considered as interchangeable: they represent the communication of verbal information (the *Message*) from a *Speaker* to an *Addressee*. This is precisely what the *Statement* frame encodes. We will use this example in the next two sections, in which we discuss our two projects in more detail.

## 2 Using an Off-the-Shelf FrameNet Parser

Our first application is SINNET, a system that extracts a social network from narrative text. It uses

the notion of a *social event* (Agarwal et al., 2010), a particular kind of event which involves (at least) two people such that at least one of them is aware of the other person. If only one person is aware of the event, we call it *Observation (OBS)*: for example, someone is talking about someone else in their absence. If both people are aware of the event, we call it *Interaction (INR)*: for example, one person is telling the other a story. Our claim is that links in social networks are in fact made up of social events: OBS social events give rise to one-way links, and INR social events to two-way links. For more information, see (Agarwal and Rambow, 2010; Agarwal et al., 2013a; Agarwal et al., 2013b).

From an NLP point of view, we have a difficult cluster of phenomena: we have a precise definition of what we want to find, but it is based on the cognitive state of the event participants, which is almost never described explicitly in the text. Furthermore, the definitions cover a large number of diverse situations such as talking, spying, having lunch, fist fighting, or kissing. Furthermore, some semantic differences are not relevant: verbs such as *talk*, *tell*, *deny*, all have the same meaning with respect to social events. Finally, in order to decode the events in terms of social events, we need to understand the roles: if I am talking to Sudeep about Mae, Sudeep and I have an INR social event with each other, and we both have a OBS social event with Mae. Thus, this problem sounds like an excellent application for frame semantics!

We present initial results in (Agarwal et al., 2014), and summarize them here. We use Semafor (Chen et al., 2010) as a black box to obtain the semantic parse of a sentence. However, there are several problems:

- FrameNet does not yet have complete lexical coverage.
- Semafor does not produce a single semantic

representation for a sentence, as we would want in order to perform subsequent processing. Instead, it annotates separate, disconnected frame structures for each frame evoking element it finds.

- The data annotated with FrameNet consists of the example sentences as well as a comparatively small corpus. For this reason, it is not easy to use standard machine learning techniques for frame semantic parsing. As a result, the output is fairly errorful (as compared to, say, a state-of-the-art dependency parser trained on nearly a million annotated words). Errors include mislabeled frames, mislabeled frame elements, and missing frame elements.

To overcome these problems, we constructed several tree representations out of the partial annotations returned by Semafor. We then used tree kernels on these syntactic and semantic tree representations, as well as bags of words. The tree kernels can automatically identify important substructures in the syntactic and semantic trees without the need for feature engineering on our part. Our hypothesis is that the kernels can learn which parts of the semantic structures are reliable and can be used for prediction.

The tree structures are shown in Figure 1. The structure on the left (FrameForest) is created by taking all identified instances of frames, and collecting them under a common root node. The frame elements are filled in with dependency syntax. The structure on the right (FrameTree) is our attempt to create a single arborescent structure to capture the semantics of the whole sentence. Our third structure, FrameTreeProp (not shown), is derived from FrameTree by multiplying the nodes of interest up the path from their normal place to the root. This allows us to overcome problems with the limited locality of the tree kernels.

We present some results in Table 1. Comparing lines “Syntax” with “Synt\_FrameTreeProp”, we see a slight but statistically significant increase. This increase comes from using FrameNet semantics. When we look at only the semantic structures, we see that they all perform worse than syntax on its own. “BOF” is simply a bag of frames; we see that the arborescent structures outperform it, so semantic structure is useful in addition to semantic tags. “RULES” is a comprehensive set of hand-written rules we attached to frames; if frame

Model	Detection		
	P	R	F1
Syntax	0.464	0.751	0.574
RULES	0.508	0.097	0.164
BOF	0.296	0.416	0.346
FrameForest	0.331	0.594	0.425
FrameTree	0.295	0.594	0.395
FrameTreeProp	0.308	0.554	0.396
All	0.494	0.641	0.558
Synt_FrameTreeProp	0.484	0.740	<b>0.585</b>

Table 1: Results for Social Event Detection. “Syntax” is an optimized model using various syntactic representations (Agarwal and Rambow, 2010). The next five models are the novel semantic features and structures. “All” refers to the model that uses all the listed structures together. “Synt\_FrameTreeProp” is a linear combination of “Syntax” and FrameTreeProp.

semantic parsing were perfect, these rules should perform pretty well. They do in fact achieve the best precision of all our systems, but the recall is so low that overall they are not useful. We interpret this result as supporting our claim that part of the problem with using frame-semantic parsers is the high error rate.

Even though the gain so far from frame semantic parsing is small, we are encouraged by the fact that an off-the-shelf semantic parser can help at all. We are currently exploring other semantic structures we can create from the semantic parse, including structures which are dags rather than trees. We would like to point out that the combination of the parser, the creation of our semantic trees, and the training with tree kernels can be applied to any other problem that is sensitive to the meaning of text. Based on our experience, we expect to see an increase in “black box” uses of FrameNet parsing for other applications in NLP.

### 3 Extending the FrameNet Resource

FrameNet can be a useful starting point for a richer knowledge representation which is needed for a specific task. In our example, we need a representation that we can use in the WordsEye project (Coyne and Sproat, 2001), in which pictures are created automatically from text descriptions. This can be understood as providing a particular type of decompositional semantics for the input text.

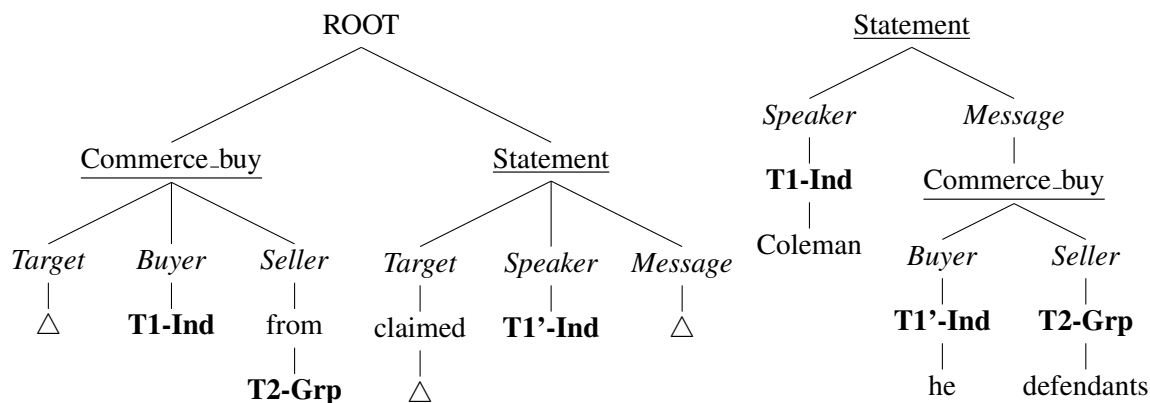


Figure 1: Semantic trees for the sentence “Coleman claimed [he]<sub>T1-Ind</sub> bought drugs from the [defendants]<sub>T2-Grp</sub>”. The tree on the left is FrameForest and the tree on the right is FrameTree.  $\triangle$  in FrameForest refers to the subtree (bought (T1-Ind) (from T2-Grp)). **Ind** refers to individual and **Grp** refers to group.

We extend FrameNet in two ways to obtain the resource we need, which we call *VigNet* (Coyne et al., 2011).

The pictures created by the WordsEye system are based on spatial arrangements (scenes) of predefined 3D models. At a low level, scenes are described by primitive spatial relations between sets of these models (*The man is in front of the woman. He is looking at her. His mouth is open.*). We would like to use WordsEye to depict scenarios, events, and actions (*John told Mary his life story*). These can be seen as complex relations between event participants.

We turn to FrameNet frames as representations for such relations. FrameNet offers a large inventory of frames, together with additional structured information about them in the form of frame relations. Most importantly, FrameNet provides example annotations illustrating the patterns in which frames are evoked and syntactic arguments are mapped to frame elements.

However, there are two main problems if we want to turn frame annotations into pictures. First, in frame annotations frame elements are only filled with text spans, not with semantic objects. Annotations are therefore restricted to individual predicate/argument structures and do not represent the meaning of a full sentence. To address this problem we essentially use FrameNet frames as an inventory of predicates in a graph-based semantic representation. We use *semantic nodes*, which are identifiers representing events and entities that fill frame elements. Frame instances then describe relations between these semantic nodes, building a

graph structure that can represent a full text fragment (including coreference). We are planning to develop parsers that convert text directly into such graph-based representations, inspired by recent work on semantic parsing (Jones et al., 2012).

Second, FrameNet frames usually describe functional relationships between frame elements, not graphical ones. To turn a frame into its graphical representation we therefore need (a) a set of graphical frames and a formal way of decomposing these frames into primitives and (b) a mechanism for relating FrameNet frames to graphical frames. Our solution is *VigNet* (Coyne et al., 2011), an extension of FrameNet. *VigNet* makes use of existing frame-to-frame relations to extend FrameNet with a number of graphical frames called *Vignettes*. Vignettes are subframes of FrameNet frames, each representing a specific way in which a frame can be realized based on the specific lexical unit or on context. For instance, a proper visualization of the INGESTION frame will depend on the INGESTOR (human vs. animals of different sizes), the INGESTIBLE (different types of foods and drinks are ingested according to different social conventions, each a different Vignette). Note however, that many FrameNet frames provide useful abstractions that allow us to use a single Vignette as a good default visualization for the entire frame. For instance, all lexical units in the STATEMENT frame can be depicted as the SPEAKER standing opposite of the ADDRESSEE with an open mouth.

A new frame-to-frame relation, called *subframe parallel*, is used to decompose a Vignette into

graphical sub-relations, which are in turn frames (either graphical primitives or other vignettes). Like any frame-to-frame relation, it maps frame elements of the source frame to frame elements of the target frame. New frame elements can also be introduced. For instance, one Vignette for INGESTION that can be used if the INGESTIBLE is a liquid contains a new frame element CONTAINER. The INGESTOR is holding the container and the liquid is in the container.

We have populated the *VigNet* resource using a number of different approaches (Coyne et al., 2012), including multiple choice questions on Amazon Mechanical Turk to define vignettes for locations (rooms), using the system itself to define locations, and a number of web-based annotation tools to define vignettes for actions.

An ongoing project is exploring the use of WordsEye and VigNet as a tool for field linguists and for language documentation and preservation. The WordsEye Linguistics Toolkit (WELT, (Ulinski et al., 2014)) makes it easy to produce pictures for field linguistic elicitation. It will also provide an environment to essentially develop language specific VigNets as models of the syntax/semantics interface and conceptual categories. This work may be relevant to other projects that aim to build non-English and multi-lingual FrameNets.

## 4 Conclusion

We have tried to motivate the claim that FrameNet provides the right layer of semantic abstraction for many NLP applications by summarizing two ongoing NLP projects at Columbia. We have also suggested that part of the problem in using FrameNet in NLP projects is the lack of a single structure that is produced, either in manual annotations, or in the output of a FrameNet parser. We suspect that research into how to construct such unified semantic representations will continue to be a major component of the use of FrameNet in NLP.

## Acknowledgments

This paper is based upon work supported in part by the NSF (grants IIS-0713548 and IIS-0904361), and by the DARPA DEFT Program. We thank our collaborators on the two projects used as examples in this extended abstract. We thank Chuck Fillmore for FrameNet.

## References

- Apoorv Agarwal and Owen Rambow. 2010. Automatic detection and classification of social events. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 1024–1034, Cambridge, MA, October. Association for Computational Linguistics.
- Apoorv Agarwal, Owen C. Rambow, and Rebecca J. Passonneau. 2010. Annotation scheme for social network extraction from text. In *Proceedings of the Fourth Linguistic Annotation Workshop*.
- Apoorv Agarwal, Anup Kotalwar, and Owen Rambow. 2013a. Automatic extraction of social networks from literary text: A case study on alice in wonderland. In *the Proceedings of the 6th International Joint Conference on Natural Language Processing (IJCNLP 2013)*.
- Apoorv Agarwal, Anup Kotalwar, Jiehan Zheng, and Owen Rambow. 2013b. Sinnet: Social interaction network extractor from text. In *Sixth International Joint Conference on Natural Language Processing*, page 33.
- Apoorv Agarwal, Sriramkumar Balasubramanian, Anup Kotalwar, Jiehan Zheng, and Owen Rambow. 2014. Frame semantic tree kernels for social network extraction from text. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, Gothenburg, Sweden.
- Desai Chen, Nathan Schneider, Dipanjan Das, and Noah A. Smith. 2010. Semafor: Frame argument resolution with log-linear models. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 264–267, Uppsala, Sweden, July. Association for Computational Linguistics.
- Bob Coyne and Richard Sproat. 2001. Wordseye: an automatic text-to-scene conversion system. In *28th annual conference on Computer graphics and interactive techniques*.
- Bob Coyne, Daniel Bauer, and Owen Rambow. 2011. Vignet: Grounding language in graphics using frame semantics. In *ACL Workshop on Relational Semantics (RELMS)*, Portland, Oregon.
- Bob Coyne, Alex Klapheke, Masoud Rouhizadeh, Richard Sproat, and Daniel Bauer. 2012. Annotation tools and knowledge representation for a text-to-scene system. In *COLING*, Mumbai, India.
- Bevan Jones, Jacob Andreas\*, Daniel Bauer\*, Karl Moritz Hermann\*, and Kevin Knight. 2012. Semantics-based machine translation with hyperedge replacement grammars. In *COLING*, Mumbai, India. \*first authorship shared.
- Morgan Ulinski, Anusha Balakrishnan, Daniel Bauer, Bob Coyne, Julia Hirschberg, and Owen Rambow. 2014. Documenting endangered languages with the wordseye linguistics tool. In *Proceedings of the ACL ComputEL workshop: The use of computational methods in the study of endangered languages*, Baltimore, MD, USA.