

Augmenting FrameNet Via PPDB

Pushpendre Rastogi¹ and Benjamin Van Durme^{1,2}

¹Center for Language and Speech Processing

²Human Language Technology Center of Excellence

Johns Hopkins University

pushpendre@jhu.edu, vandurme@cs.jhu.edu

Abstract

FrameNet is a lexico-semantic dataset that embodies the theory of frame semantics. Like other semantic databases, FrameNet is incomplete. We augment it via the paraphrase database, PPDB, and gain a three-fold increase in coverage at 65% precision.

1 Introduction

Frame semantics describes the meaning of a word in relation to real world events and entities. In frame semantics the primary unit of lexical analysis is the *frame* and the *lexical unit*. A frame aims to capture the most salient properties of a concept, situation or event. For example, the frame representing the concept of *Abandonment* contains eight attributes:¹ *Agent*, *Theme*, *Place*, *Time*, *Manner*, *Duration*, *Explanation* and *Depictive*. A *lexical unit* is a tuple of three elements: the lemma of a word, its POS tag and the associated frame.

FrameNet is large lexico-semantic dataset that contains manually annotated information including frame descriptions, frame-frame relations and frame annotated sentences. It has been used build to frame semantic parsers, which are systems that can analyze a sentence and annotate its words with the frames that they evoke and the corresponding frame elements. The task of frame semantic parsing was introduced by Gildea and Jurafsky (2002) and later it matured into a community-wide shared task (Baker et al., 2007), with CMU’s SEMAFOR system being the current state-of-the-art parser (Das et al., 2013).

Common to rich, manually constructed semantic resources, the coverage of FrameNet across its

targetted language (English) is incomplete. State-of-the-art frame semantic parsers thus employ various heuristics to identify the frame evoked by out-of-vocabulary items (OOVs) at test-time.² For instance, an OOV if present in WordNet might be aligned to frame(s) assigned to in-vocabulary items in shared synsets (see the work by Ferrández et al. (2010) and the related works section therein). In this work we take a different approach and attempt to directly increase the coverage of the FrameNet corpus by automatically expanding the collection of training examples via PPDB, The Paraphrase Database (Ganitkevitch et al., 2013).

In Section 2 we analyze FrameNet and comment on the sparsity in its different parts. In Section 3 we describe PPDB, and how it was used to augment FrameNet. We present our evaluation experiments and results in the latter half of the section followed by conclusions.

2 FrameNet Coverage

FrameNet is a rich semantic resource, yet currently lacks complete coverage of the language. In the following we give examples of this incompleteness, in particular the OOV issue that we will focus on in latter sections.

Frames A frame represents an event, a situation or a real life concept; FrameNet version 1.5 contains 1,019 such frames. These thousand frames do not cover all possible situations that we might encounter. For example, FrameNet does not have a frame for the activity of *Programming* even though it has frames for *Creating*, *Text_Creation*, etc. The situa-

¹An attribute of a frame is also called a *Frame Element*.

²For example the *Abandonment* frame lacks *jettison* as one of its lexical units, and further, that word is not listed as a lexical unit in FrameNet v1.5, making *jettison* an OOV.

tion of writing a computer program is stereotypical and attributes that a reader might associate with such an activity are: agent (who wrote the program), language (the programming language used) and function (the program's purpose).

Further, FrameNet's granularity is at times uneven. For example, the `Explosion` frame and the `BecomeTriggered` frames do not have agentive attributes, instead there exist separate frames `DetonateExplosive` and `Triggering` which have the attributes `Agent` and `Actor` respectively. This suggests a pattern that events which are frequently described without an agent are assigned their own frames. However, there is no `Burial` frame which focuses on the event corresponding to frame of `Burying`, which itself focuses on the `Actor`.

This difference in granularity could be resolved by either making distinctions more evenly fine-grained: trying to automatically inducing new frames; or by making things more evenly-coarse grained: automatically merging existing frames that are deemed similar. Researchers have explored methods for automatically learning frames and on learning collocations of frames to their syntactic dependent phrases. Recent examples include using either a Bayesian topic model to learn clusters of words (O' Connor, 2012; Materna, 2012), or attempting to learn symbolic concepts and attributes from dictionary definitions of words (Orfan and Allen, 2013).

Frame-Frame Relations FrameNet encodes certain types of correlations between situations and events by adding defeasible typed-relations between frames encoding pairwise dependencies.

There are eight types of frame-frame relations: `Inherits_from`, `Perspective_on`, `Precedes`, `Subframe_of`, `See_also`, `Uses`, `Is_Inchoative_of`, and `Is_Causative_of`.³ For example the frame `Being_Born` is related to `Death` through the relation `Is_Preceded_By`. Such common-sense knowledge of event-event relationships would be of significant utility to general AI, however it is a large space to fill: with 1,019 frames and 8 binary relations there is a large upper bound on the number of total possible

³Five frame-frame relations also have an antonym relation: `Is_Used_by`, `Is_Inherited_by`, `Is_Perspectivized_in`, `Has_Subframe(s)`, `Is_Preceded_by`, however an antonym relation does not add any extra information over its corresponding relation.

relation pairs, even if not considering the previous issue of incomplete frame coverage. For example, the `Experience_bodily_harm` and `Hostile_encounter` frames are not related through the `Is_Causative_of` relation, even though it is reasonable to expect that a hostile encounter would result in bodily harm.⁴ Though researchers have used FrameNet relations for tasks such as recognizing textual entailment (RTE) (Aharon et al., 2010) and for text understanding (Fillmore and Baker, 2001), to the best of our knowledge there has been no work on automatically extending frame-frame relations.

Frame Annotated Sentences FrameNet contains annotated sentences providing examples of: lexical units, frames those lexical units evoked, and frame elements present in the sentence (along with additional information). These annotated sentences can be divided into two types based on whether all the frame evoking words were marked as targets or not.

The first type, which we call lexicographic, contains sentences with a single target per sentence. The second type, called fulltext, contains sentences that have been annotated more completely and they contain multiple targets per sentence. There are 4,026 fulltext sentences containing 23,921 targets. This data has proved to be useful for lexico-semantic tasks like RTE and paraphrasing e.g. (Aharon et al., 2010; Coyne and Rambow, 2009). As compared to PropBank (Palmer et al., 2005), which annotated all predicates occurring within a collection of pre-existing documents, FrameNet provides *examples*, but not a corpus that allows for directly estimating relative frequencies.

Frame-Lemma Mappings As said earlier, *lexical units* are tuples of the lemma form of a word, its POS-tag and its associated frame. One component of FrameNet is its information about which words/lemmas prompt a particular frame. An annotated word that evokes a frame in a sentence is referred to as a *Target*. There are two areas where these mappings could be incomplete: (1) lemmas contained within FrameNet may have alternate senses such that they should be placed in more Frames (or related: a currently missing frame might then give rise to another sense of

⁴*Reasonable* highlights the issue that we would optimally like to know things that are even just possible/not-too-unlikely, even if not strictly entailed.

such a lemma); and (2) lemmas from the language may not be in FrameNet in any form. Most research on mitigating this limitation involves mapping FrameNet’s frames to WordNet’s synsets.⁵ Fossati et al. (2013) explored the feasibility of crowdsourcing FrameNet coverage, using the distributed manual labor of Mechanical Turk to complete the lemma coverage.

3 Augmenting FrameNet with PPDB

In order to expand the coverage of FrameNet, we performed an initial study on the use of a new broad-coverage lexical-semantic resource, PPDB, to first add new lemmas as potential triggers for a frame, and then automatically rewrite existing example sentences with these new triggers. The eventual goal would be to enable any existing FrameNet semantic parser to simply retrain on this expanded dataset, rather than needing to encode methods for dynamic OOV-resolution at test-time (such as employed by SEMAFOR).

PPDB Ganitkevitch et al. (2013) released a large collection of lexical, phrasal and syntactic paraphrases⁶ collectively called PPDB. We used the lexical rules in PPDB to find potential paraphrases of target words of frame annotated sentences. A lexical rule in PPDB looks like the following:

```
[VB] ||| help ||| assist |||  
p(e|f)=2.832, p(f|e)=1.551, ...
```

This rule conveys that the log-probability that *help* would be paraphrased by the word *assist* is -2.832 but the log probability of *assist* being paraphrased as *help* is -1.551.⁷ Ganitkevitch et al. (2013) released quality-sorted subsets of the full (large) collection, varying in size from S to XXXL by applying thresholds over a linear combination of the feature values to prune away low quality paraphrases. They verified that the average human judgement score of randomly sampled paraphrases from smaller sized collections was higher than the

⁵It is worth noting that substituting a larger automatically derived WordNet (as derived in Snow et al. (2004)) could improve the recall of some of the methods which automatically learn a mapping from FrameNet frames to WordNet synsets.

⁶*Lexical*: Two words with the same meaning; *phrasal*: two strings of words with the same meaning; *syntactic*: strings of surface words and non-terminal categories that have the same meaning. These strings are templates with the non-terminals serving the role of constraints over what can go in the blanks.

⁷See complete list at <http://github.com/jweese/thrax/wiki/Feature-functions>.

average human judgement score of a random sample from a larger collection.

Approach We used the lexical rules sans features along with a 5-gram Kneser-Ney smoothed language model trained using KenLM (Heafield et al., 2013) on the raw English sequence of Annotated Gigaword (Napoles et al., 2012) to paraphrase the fulltext frame annotated sentences of FrameNet. We used a combination of the WordNet morphological analyzer and Morpha⁸ for lemmatization and Morphg⁹ for generation.

Evaluation We present our evaluation of the quantity and quality of generated paraphrases in this section. Note that we did not use syntactic reordering to generate the paraphrases. Also we paraphrased the frame evoking targets individually i.e. we did not consider combinations of paraphrases of individual targets to be a new paraphrase of a sentence and we ignored those frame evoking targets that contained multiple words.¹⁰

With the above mentioned constraints we conducted the following experiments with different sizes of PPDB. In Experiment 1 we generated a set of candidate paraphrases for every target word in a sentence by directly querying that word and its dictionary form in PPDB. In Experiment 2 we first enlarged the set of lexical units mapped to a frame by merging lexical units of frames that were related to the target word’s frame through either of the following relations: *Is.Perspectivized.In*, *Is.Inherited.By*, *Has.Subframe(s)*. For example, if frame A has a subframe B then lexical units mapped to A can evoke B. We then queried PPDB to gather paraphrases for all the lexical units collected so far. This experiment simulates the situation where a frame has been mapped to a set of words, e.g. synsets in WordNet, so that every word in that larger set is a paraphrase of any word that evokes a frame. This procedure increases the average number of paraphrases mapped to a frame and we present those averages in Table 1.

For both these experiments we also calculated the incremental benefit of PPDB over WordNet by

⁸<http://ilexir.co.uk/applications/rasp/download>

⁹http://cl.naist.jp/~eric-n/ubuntu-nlp/pool/hardy/english/morph_0.0.20030918-2nlp1~0hardy1.tar.gz

¹⁰Among fulltext sentences less than 3% of targets contained multiple tokens.

Database	Lexical Unit/Frame
Framenet	20.24
PPDB S	23.15
PPDB M	32.00
PPDB L	74.08
PPDB XL	214.99

Table 1: Average count of lexical units per frame for different sizes of PPDB in experiment 2.

The General Assembly should set aside money for a new state health lab , millions of doses of antiviral drugs and a fund to help meet basic needs after a disaster , a legislative panel recommended Thursday .
1: The General Assembly should set aside cash ...
2: The General Assembly should set aside fund ...
1: The General Assembly should set aside dough ...
3: The General Assembly should set aside silver ...

Table 2: Examples and their judgements, with the last being debatable.

filtering out paraphrases that could have been retrieved as synonyms¹¹ from WordNet v3.0. The results of these experiments are in Table 3.

To evaluate the quality of our additional output over WordNet we assigned one of the following labels to 25 paraphrase sets generated at the end of Experiment 1b¹²: 1, the paraphrase (a) invoked the correct frame and (b) was grammatical; or 2, only (a) held; or 3, (a) did not hold. Table 4 presents aggregates over the labels.

PPDB	1a	1b	2a	2b
S	4,582	2,574	1,064,926	1,022,533
M	15,459	9,752	1,314,169	1,263,087
L	73,763	55,517	2,417,760	2,347,656
XL	340,406	283,126	-	-

Table 3: The total number of paraphrases generated for the 23,226 input targets versus different sizes of PPDB. The paraphrase count excludes the input. Column 1a and 2a represent unfiltered paraphrases as opposed to 1b and 2b where they have been filtered using WordNet v3.0.

4 Discussion And Conclusion

We presented initial experiments on using PPDB to automatically expand FrameNet through paraphrastic re-writing. We found that over a sample of 25 target words the top three paraphrases produced by PPDB XL evoked the correct frame and were grammatical 65% of the time.¹³ However,

¹¹Two lemmas that appear in the same synset at least once are synonyms in our experiments.

¹²Experiment 2 generated significantly more candidates; here we consider only the potential scope of expansion and rely on Experiment 1 to gauge the likely paraphrase quality.

¹³We have released the generated corpus as well as the manual annotations at cs.jhu.edu/~prastog3/res/fnpdb.html

PPDB Size	1	2	3	%(1+2)	%(1)
S	0	0	0	-	-
M	6	1	2	77.77	66.67
L	27	15	11	86.25	50.94
L rank 3	23	12	7	83.33	54.76
XL	110	85	50	79.60	44.89
XL rank 3	47	16	9	87.5	65.27
XL rank 5	69	28	13	88.18	62.72
XL rank 10	105	52	32	83.07	55.55

Table 4: Average quality of all paraphrases for 25 random sentences. Rows marked *A rank B* convey that we used PPDB of size *A* and kept only the top *B* sentences after sorting them by their language model score. Column %(1) indicates the percentage of output which was grammatical and evoked the correct frame. Column %(1+2) represents the output that evoked the correct frame.

work remains in recognizing the contexts in which a paraphrase is appropriately applied, and in improving the quality of PPDB itself.

Upon error analysis, we found two major reasons for ungrammaticality of lexical paraphrases. First: within FrameNet some sentences will have a single token annotated as trigger, when in fact it is part of a multi-word expression. For example, it was grammatically infelicitous to replace *part* by *portion* in the expression *part of the answer*. The other major source of error was the inaccuracy in PPDB itself. We found that for a large number of cases when PPDB XL did not have a high number of paraphrases the paraphrases were wrong (e.g., PPDB XL had only 2 paraphrases for the words *lab* and *millions*.)

Going forward we aim to increase the precision of our paraphrases and our ability to recognize their appropriate contexts for application. Further, we wish to augment additional resources in a similar way, for example PropBank or the ACE corpus (Walker et al., 2006). We should be able to increase the precision by using the paraphrase probability features of a PPDB rule and by using better language models with lower perplexity than n-grams e.g. recurrent neural net based language models. Improving the accuracy of PPDB, especially in the large settings, would be another focus area. Also, we would use Amazon Mechanical Turk to evaluate the quality of a larger set of paraphrases to make our evaluation robust and so that we can evaluate the efficacy of our second experiment.

Acknowledgments This material is based on research sponsored by the NSF under grant IIS-1249516 and DARPA under agreement number FA8750-13-2-0017.

References

- Roni Ben Aharon, Idan Szpektor, and Ido Dagan. 2010. Generating entailment rules from framenet. In *Proceedings of the ACL 2010 Conference Short Papers*, pages 241–246.
- Collin Baker, Michael Ellsworth, and Katrin Erk. 2007. Semeval’07 task 19: frame semantic structure extraction. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, pages 99–104. ACL.
- Bob Coyne and Owen Rambow. 2009. Lexpar: A freely available english paraphrase lexicon automatically extracted from framenet. *2012 IEEE Sixth International Conference on Semantic Computing*, pages 53–58.
- Dipanjan Das, Desai Chen, André F. T. Martins, Nathan Schneider, and Noah A. Smith. 2013. Frame-semantic parsing. *Computational Linguistics*, 40(1):9–56.
- Oscar Ferrández, Michael Ellsworth, Rafael Munoz, and Collin F Baker. 2010. Aligning framenet and wordnet based on semantic neighborhoods. In *LREC*, volume 10, pages 310–314.
- Charles J Fillmore and Collin F Baker. 2001. Frame semantics for text understanding. In *Proceedings of WordNet and Other Lexical Resources Workshop, NAACL*.
- Marco Fossati, Claudio Giuliano, and Sara Tonelli. 2013. Outsourcing framenet to the crowd. In *Proceedings of the 51st Annual Meeting of the ACL*, pages 742–747.
- Juri Ganitkevitch, Benjamin Van Durme, and Chris Callison-Burch. 2013. PPDB: The paraphrase database. In *Proceedings of NAACL-HLT*, pages 758–764, Atlanta, Georgia, June. ACL.
- Daniel Gildea and Daniel Jurafsky. 2002. Automatic labeling of semantic roles. *Computational linguistics*, 28(3):245–288.
- Kenneth Heafield, Ivan Pouzyrevsky, Jonathan H. Clark, and Philipp Koehn. 2013. Scalable modified Kneser-Ney language model estimation. In *Proceedings of the 51st Annual Meeting of the ACL*, Sofia, Bulgaria.
- Jiří Materna. 2012. Lda-frames: An unsupervised approach to generating semantic frames. In *Computational Linguistics and Intelligent Text Processing*, volume 7181 of *Lecture Notes in Computer Science*, pages 376–387. Springer Berlin Heidelberg.
- Courtney Napoles, Matthew Gormley, and Benjamin Van Durme. 2012. Annotated gigaword. In *Proceedings of the Joint Workshop on Automatic Knowledge Base Construction and Web-scale Knowledge Extraction*, pages 95–100. ACL.
- Brendan O’ Connor. 2012. Learning frames from text with an unsupervised latent variable model. In *Technical Report*. Carnegie Mellon University.
- Jansen Orfan and James Allen. 2013. Toward learning high-level semantic frames from definitions. In *Proceedings of the Second Annual Conference on Advances in Cognitive Systems*, volume 125.
- Martha Palmer, Daniel Gildea, and Paul Kingsbury. 2005. The proposition bank: An annotated corpus of semantic roles. *Computational Linguistics*, 31(1):71–106.
- Rion Snow, Daniel Jurafsky, and Andrew Y Ng. 2004. Learning syntactic patterns for automatic hypernym discovery. In *NIPS*, volume 17, pages 1297–1304.
- Christopher Walker, Stephanie Strassel, Julie Medero, and Kazuaki Maeda. 2006. Ace 2005 multilingual training corpus. *Linguistic Data Consortium, Philadelphia*.