# A scaleable automated quality assurance technique for semantic representations and proposition banks

**K. Bretonnel Cohen**
Computational Bioscience Program
U. of Colorado School of Medicine
Department of Linguistics
University of Colorado at Boulder
kevin.cohen@gmail.com

**Lawrence E. Hunter**
Computational Bioscience Program
U. of Colorado School of Medicine
larry.hunter@ucdenver.edu

**Martha Palmer**
Department of Linguistics
University of Colorado at Boulder
martha.palmer@colorado.edu

## Abstract

This paper presents an evaluation of an automated quality assurance technique for a type of semantic representation known as a predicate argument structure. These representations are crucial to the development of an important class of corpus known as a proposition bank. Previous work (Cohen and Hunter, 2006) proposed and tested an analytical technique based on a simple discovery procedure inspired by classic structural linguistic methodology. Cohen and Hunter applied the technique manually to a small set of representations. Here we test the feasibility of automating the technique, as well as the ability of the technique to scale to a set of semantic representations and to a corpus many times larger than that used by Cohen and Hunter. We conclude that the technique is completely automatable, uncovers missing sense distinctions and other bad semantic representations, and does scale well, performing at an accuracy of 69% for identifying bad representations. We also report on the implications of our findings for the correctness of the semantic representations in PropBank.

## 1 Introduction

It has recently been suggested that in addition to more, bigger, and better resources, we need a science of creating them (Palmer et al., Download date December 17 2010).

The corpus linguistics community has arguably been developing at least a nascent science of annotation for years, represented by publications such as (Leech, 1993; Ide and Brew, 2000; Wynne, 2005; Cohen et al., 2005a; Cohen et al., 2005b) that address architectural, sampling, and procedural issues, as well as publications such as (Hripcsak and Rothschild, 2005; Artstein and Poesio, 2008) that address issues in inter-annotator agreement. However, there is not yet a significant body of work on the subject of quality assurance for corpora, or for that matter, for many other types of linguistic resources. (Meyers et al., 2004) describe three error-checking measures used in the construction of NomBank, and the use of inter-annotator agreement as a quality control measure for corpus construction is discussed at some length in (Marcus et al., 1993; Palmer et al., 2005). However, discussion of quality control for corpora is otherwise limited or nonexistent.

With the exception of the inter-annotator-agreement-oriented work mentioned above, none of this work is quantitative. This is a problem if our goal is the development of a true science of annotation.

Work on quality assurance for computational lexical resources other than ontologies is especially lacking. However, the body of work on quality assurance for ontologies (Kohler et al., 2006; Ceusters et al., 2004; Cimino et al., 2003; Cimino, 1998; Cimino, 2001; Ogren et al., 2004) is worth considering in the context of this paper. One common theme in that work is that even manually curated lexical resources contain some percentage of errors.

The small size of the numbers of errors uncovered in some of these studies should not be taken as a significance-reducing factor for the development of quality assurance measures for lexical resources—

*Proceedings of the Fifth Law Workshop (LAW V)*, pages 82–91,
Portland, Oregon, 23-24 June 2011. ©2011 Association for Computational Linguistics

rather, the opposite: as lexical resources become larger, it becomes correspondingly more difficult to locate errors in them. Finding problems in a very errorful resource is easy; finding them in a mostly correct resource is an entirely different challenge.

We present here an evaluation of a methodology for quality assurance for a particular type of lexical resource: the class of semantic representation known as a predicate argument structure (PAS). Predicate argument structures are important in the context of resource development in part because they are the fundamental annotation target of the class of corpus known as a proposition bank. Much of the significance claim for this work comes from the significance of proposition banks themselves in recent research on natural language processing and computational lexical semantics. The impact of proposition banks on work in these fields is suggested by the large number of citations of just the three publications (Kingsbury and Palmer, 2002; Kingsbury et al., 2002; Palmer et al., 2005)—at the time of writing, 290, 220, and 567, respectively. Additional indications of the impact of PropBank on the field of natural language processing include its use as the data source for two shared tasks ((Carreras and Màrquez, 2005)).

The methodology consists of looking for arguments that never coöccur with each other. In structural linguistics, this property of non-coöccurrence is known as *complementary distribution*. Complementary distribution occurs when two linguistic elements never occur in the same environment. In this case, the environment is defined as any sentence containing a given predicate. Earlier work showed a proof-of-concept application to a small set of rolesets (defined below) representing the potential PAS of 34 biomedical predicates (Cohen and Hunter 2006). The only inputs to the method are a set of rolesets and a corpus annotated with respect to those rolesets. Here, we evaluate the ability of the technique to scale to a set of semantic representations 137 times larger (4,654 in PropBank versus 34 in Cohen and Hunter's pilot project) and to a corpus about 1500 times larger (1M words in PropBank versus about 680 in Cohen and Hunter's pilot project) than that considered in previous work. We also use a set of independent judges to assess the technique, where in the earlier work, the results were only assessed by one of the authors.

Novel aspects of the current study include:

- Investigating the feasibility of automating the previously manual process
- Scaling up the size of the set of semantic representations evaluated
- Scaling up the size of the corpus against which the representations are evaluated
- Using independent judges to assess the predictions of the method

### 1.1 Definitions

For clarity, we define the terms *roleset, frame file,* and *predicate* here. A *roleset* is a 2-tuple of a sense for a predicate, identified by a combination of a lemma and a number—e.g., *love.01*—and a set of individual thematic roles for that predicate—e.g., *Arg0 lover* and *Arg1 loved.* A *frame file* is the set of all rolesets for a single lemma—e.g., for *love,* the rolesets are *love.01* (the sense whose antonym is *hate*) and *love.02*, the "semi-modal" sense in *whether it be melancholy or gay, I love to recall it* (Austen, 1811). Finally, we refer to sense-labelled predicates (e.g. *love.01*) as *predicates* in the remainder of the paper.

PropBank rolesets contain two sorts of thematic roles: (core) arguments and (non-core) adjuncts. Arguments are considered central to the semantics of the predicate, e.g. the Arg0 *lover* of *love.01*. Adjuncts are not central to the semantics and can occur with many predicates; examples of adjuncts include negation, temporal expressions, and locations.

In this paper, the *arity* of a roleset is determined by its count of arguments, disregarding adjuncts.

### 1.2 The relationship between observed argument distributions and various characteristics of the corpus

This work is predicated on the hypothesis that argument distributions are affected by goodness of the fit between the argument set and the actual semantics of the predicate. However, the argument distributions that are observed in a specific data set can be affected by other factors, as well. These include at least:

- Inflectional and derivational forms attested in the corpus

- Sublanguage characteristics
- Incidence of the predicate in the corpus

A likely cause of derivational effects on observed distributions is nominalization processes. Nominalization is well known for being associated with the omission of agentive arguments (Koptjevskaja-Tamm, 1993). A genre in which nominalization is frequent might therefore show fewer coöccurrences of Arg0s with other arguments. Since PropBank does not include annotations of nominalizations, this phenomenon had no effect on this particular study.

Sublanguage characteristics might also affect observed distributions. The sublanguage of recipes has been noted to exhibit rampant deletions of definite object noun phrases both in French and in English, as has the sublanguage of technical manuals in English. (Neither of these sublanguages have been noted to occur in the PropBank corpus. The sublanguage of stock reports, however, presumably does occur in the corpus; this sublanguage *has* been noted to exhibit distributional subtleties of predicates and their arguments that might be relevant to the accuracy of the semantic representations in PropBank, but the distributional facts do not seem to include variability in argument coöccurrence so much as patterns of argument/predicate coöccurrence (Kittredge, 1982).)

Finally, incidence of the predicate in the corpus could affect the observed distribution, and in particular, the range of argument coöccurrences that are attested: the lower the number of observations of a predicate, the lower the chance of observing any two arguments together, and as the number of arguments in a roleset increases, the higher the chance of failing to see any pair together. That is, for a roleset with an arity of three and an incidence of *n* occurrences in a corpus, the likelihood of never seeing any two of the three arguments together is much lower than for a roleset with an arity of six and an incidence of *n* occurrences in the corpus. The number of observations required in order to be able to draw conclusions about the observed argument distributions with some degree of confidence is an empirical question; prior work (Cohen and Hunter 2006) suggests that as few as ten tokens can be sufficient to uncover erroneous representations for rolesets with an arity of four or less, although that number of observations of one roleset with an arity of four showed multiple non-coöccurring arguments that were not obviously indicative of problems with the representation (i.e., a false positive finding).

Besides the effects of these aspects of the corpus contents on the observed distributions, there are also a number of theoretical and practical issues in the design and construction of the corpus (as distinct from the rolesets, or the distributional characteristics of the contents) which have nontrivial implications for the methodology being evaluated here. In particular, the implications of the argument/adjunct distinction, of the choice of syntactic representation, and of annotation errors are all discussed in Section 4. Note that we are aware that corpus-based studies generally yield new lexical items and usages any time a new corpus is introduced, so we do not make the naive assumption that PropBank will give complete coverage of all coöccurring arguments, and in fact our evaluation procedure took this into account explicitly, as described in Section 2.3.

## 2 Materials and Methods

### 2.1 Materials

We used Rev. 1.0 of the PropBank I corpus, and the associated framesets in the `frames` directory.

### 2.2 Methods

#### 2.2.1 Determining the distribution of arguments for a roleset

In determining the possible coöccurring argument pairs for a roleset, we considered only arguments, not adjuncts. As we discuss in Section 4.1, this is a non-trivial decision with potential implications for the ability of the algorithm to detect problematic representations in general, and with implications for PropBank in particular. The rationale behind the choice to consider only arguments is that our goal is to evaluate the representation of the semantics of the predicates, and that by definition, the PropBank arguments are essential to defining that semantics, while by definition, the adjuncts are not.

In the first processing step, for each roleset, we used the corresponding framefile as input and generated a look-up table of the possible argument pairs for that predicate. For example, the predicate *post.01* has the three arguments *Arg0, Arg1,* and

*Arg2*; we generated the set {<Arg0, Arg1>, <Arg0, Arg2>, <Arg1, Arg2>} for it.

In the second processing step, we iterated over all annotations in the PropBank corpus, and for each token of each predicate, we extracted the complete set of arguments that occurred in association with that token. We then constructed the set of coöccurring arguments for that annotation, and used it to increment the counts of each potential argument pair for the predicate in question. For example, the PropBank annotation for *Oils and fats also did well, posting a 5.3% sales increase* (`wsj/06/wsj_0663.mrg`) contains an Arg0 and an Arg1, so we incremented the count for that argument pair by 1; it contains no other argument pairs, so we did not increment the counts for <Arg0, Arg2> or <Arg1, Arg2>.

The output of this step was a table with the count of occurrence of every potential pair of arguments for every roleset; members of pairs whose count was zero were then output as arguments in complementary distribution. For example, for *post.01,* the pairs <Arg0, Arg2> and <Arg1, Arg2> never occurred, even as traces, so the arguments Arg0 and Arg2 are in complementary distribution for this predicate, as are the arguments Arg1 and Arg2.

To manipulate the data, we used Scott Cotton's Java API, with some extensions, which we documented in the API's Javadoc.

### 2.3 Determining the goodness of rolesets exhibiting complementary distribution

In (Cohen and Hunter, 2006), determinations of the goodness of rolesets were made by pointing out the distributional data to the corpus creators, showing them the corresponding data, and reaching consensus with them about the appropriate fixes to the representations. For this larger-scale project, one of the goals was to obtain goodness judgements from completely independent third parties.

Towards that end, two judges with experience in working with PropBank were assigned to judge the predictions of the algorithm. Judge 1 had two years of experience, and Judge 2 had four years of experience. The judges were then given a typology of classification to assign to the predicates: good, bad, and conditionally bad. The definitions of these categories, with the topology of the typology, were:

- **Good:** This label is assigned to predicates that the algorithm predicted to have bad representations, but that are actually good. They are false positives for the method.
- **Not good:** (This label was not actually assigned, but rather was used to group the following two categories.)
  - **Bad:** This label is assigned to predicates that the algorithm predicted to have bad representations and that the judges agreed were bad. They are true positives for the method.
  - **Conditionally bad:** This label is assigned to predicates that the algorithm predicted to have bad representations and that the judges agreed were bad based on the evidence available in PropBank, but that the judges thought might be good based on native speaker intiution or other evidence. In all of these cases, the judges did suggest changes to the representations, and they were counted as not good, per the typology, and are also true positives.

Judges were also asked to indicate whether bad representations should be fixed by splitting predicates into more word senses, or by eliminating or merging one or more arguments.

We then took the lists of all predicted bad predicates that appeared at least 50, 100, or 200 times in the PropBank corpus. These were combined into a single list of 107 predicates and randomized. The judges then split the list into halves, and each judge examined half of the list. Additionally, 31 predicates, or 29% of the data set, were randomly selected for double annotation by both judges to assess interjudge agreement. Judges were shown both the predicates themselves and the sets of non-coöccurring arguments for each predicate.

## 3 Results

### 3.1 Accuracy

The overall results were that out of 107 predicates, 33 were judged GOOD, i.e. were false positives. 44 were judged BAD and 30 were judged CONDITIONAL, i.e. were true positives. This yields a ratio of 2.24 of true positives to false positives: the pro-

Table 1: Ratios of BAD plus CONDITIONAL to GOOD for the pooled judgements as broken down by arity

| Arity | Ratio |
|---|---|
| 3 | 1.29 |
| 4 | 1.47 |
| 5 | 4.0 |
| 6 | 8.0 |
| 7 | None found |

cedure returns about two true positives for every one false positive. Expressed in terms of accuracy, this corresponds to 69% for correctly labelling true positives.

We broke down the data by (1) arity of the roleset, and (2) minimum number of observations of a role set. This allowed us to test whether predictive power decreased as arity increased, and to test the dependency of the algorithm on the minimum number of observations; we suspected that it might be less accurate the fewer the number of observations.

Table 1 shows the ratios of true positives to false positives, broken down by arity. The data confirms that the algorithm is effective at finding bad representations, with the number of true positives outnumbering the number of false positives at every arity. This data is also important because it allows us to test a hypothesis: is it the case that predictive power becomes worse as arity increases? As the table shows, the ratio of true positives to false positives actually increases as the arity of the predicate increases. Therefore, the data is consistent with the hypothesis that not only does the predictive power of the algorithm not lessen as arity increases, but rather it actually becomes greater.

Table 2 shows the ratios of true positives to false positives again, this time broken down by minimum number of occurrences of the predicates. Again, the data confirms that the algorithm is effective at finding bad representations—it returns more bad representations than good representations at every level of minimum number of observations. This data is also important because it allows us to test the hypothesis of whether or not predictive power of the algorithm decreases with the minimum number of observations. As we hypothesized, it does show that the predictive power decreases as the minimum number of observations decreases, with the ratio of true positives to false positives dropping from 2.63 with a minimum of 200 or 100 observations to 1.88 with a minimum of 50 observations. However, the ratio of true positives to false positives remains close to 2:1 at every level.

Table 2: Ratios of BAD plus CONDITIONAL to GOOD for the pooled judgements as broken down by minimum number of observations

| | ratio |
|---|---|
| Minimum 50 | 1.88 |
| Minimum 100 | 2.63 |
| Minimum 200 | 2.63 |

### 3.2 Suggested fixes to the representations

Of the 74 true positives, the judges felt that 17 of the bad representations should be fixed by splitting the predicate into multiple senses. For the 57 remaining true positives, the judges felt that an argument should be removed from the representation or converted to an adjunct. This demonstrates that the method is applicable both to the problem of revealing missing sense distinctions and to the problem of identifying bad arguments.

### 3.3 Scalability

The running time was less than one and a half minutes for all 4,654 rolesets on the 1-million-word corpus.

### 3.4 Inter-judge agreement

A subset of 31 predicates was double-annotated by the two judges to examine inter-judge agreement. The judges then examined the cases on which they initially disagreed, and came to a consensus where possible. Initially, the judges agreed in 63.3% of the cases, which is above chance but not the 80% agreement that we would like to see. The judges then went through a reconciliation process. They were able to come to a consensus in all cases.

### 3.5 Putting the results in context

To help put these results in context, we give here the distribution of arities in the PropBank rolesets and the minimum number of observations of each in the PropBank corpus.

Table 3: Distribution of arities by percentage and by count in the 4,654 PropBank rolesets.

| Arity | percentage (count) |
|---|---|
| 0 | 0.28% (13) |
| 1 (Arg0) | 155 |
| 1 (Arg1) | 146 |
| 1 (all) | 6.5% (301) |
| 2 | 45.14% (2,101) |
| 3 | 37.02% (1,723) |
| 4 | 7.05% (328) |
| 5 | 3.5% (163) |
| 6 | 0.5% (24) |
| 7 | 0.0002% (1) |
| Total | 100% (4,654) |

Table 3 shows the distribution of arities in the PropBank rolesets. It distinguishes between non-ergatives and ergatives (although for the purpose of calculating percentages, they are combined into one single-arity group). The mode is an arity of 2: 45.14% of all rolesets (2,101/4,654) have an arity of 2. 3 is a close second, with 37.02% (1,723/4,654). (The single roleset with an arity of seven is *notch.02*, with a gloss of "move incrementally.")

Table 4 gives summary statistics for the occurrence of complementary distribution, showing the distribution of rolesets in which there were at least one argument pair in complementary distribution and of the total number of argument pairs in complementary distribution. Since (as noted in Section 1.2) the incidence of a predicate has a potential effect on the incidence of argument pairs in apparent complementary distribution, we display the counts separately for four cut-offs for the minimum number of observations of the predicate: 200, 100, 50, and 10.

To further explicate the operation of the discovery procedure, we give here some examples of rolesets that were found to have arguments in complementary distribution.

### 3.5.1 *accept.01*

*Accept.01* is the only roleset for the lemma *accept*. Its sense is *take willingly*. It has four arguments:

- Arg0 acceptor
- Arg1 thing accepted
- Arg2 accepted-from
- Arg3 attribute

Table 4: Summary statistics: counts of predicates with at least one argument pair in complementary distribution and of total argument pairs in complementary distribution for four different minimum numbers of observations of the predicates.

| Minimum observations | Predicates | Argument pairs |
|---|---|---|
| 200 | 29 | 69 |
| 100 | 58 | 125 |
| 50 | 107 | 268 |
| 10 | 328 | 882 |

The predicate occurs 149 times in the corpus. The algorithm found Arg2 and Arg3 to be in complementary distribution.

Manual investigation showed the following distributional characteristics for the predicate and its arguments:

- (Arg0 or Arg1) and Arg2: 5 tokens
- (Arg0 or Arg1) and Arg3: 8 tokens
- Arg2 with neither Arg0 nor Arg1: 0 tokens
- Arg3 with neither Arg0 nor Arg1: 0 tokens
- Arg0 or Arg1 with neither Arg2 nor Arg 3: 136 tokens

Examination of the 5 tokens in which Arg2 coöccurred with Arg0 or Arg1 and the 8 tokens in which Arg3 coöccurred with Arg0 or Arg1 suggested an explanation for the complementary distribution of arguments Arg2 and Arg3. When Arg2 appeared, the sense of the verb seemed to be one of physical transfer: Arg2 coöccurred with Arg1s like *substantial gifts* (wsj_0051.mrg) and *a $3 million payment* (wsj_2071.mrg). In contrast, when Arg3 appeared, the sense was not one of physical transfer, but of some more metaphorical sense—Arg3 coöccurred with Arg1s like *the war* (wsj_0946.mrg) and *Friday's dizzying 190-point plunge* (wsj_2276.mrg). There is no *accept.02*; creating one with a 3-argument roleset including the current Arg3 seems warranted. Keeping the Arg3 for *accept.01* might be warranted, as well, but probably as an adjunct (to account for usages like *John accepted it as a gift.*)

### 3.5.2 *affect.01*

*Affect.01* is one of two senses for the lemma *affect.* Its sense is *have an effect on.* It has three arguments:

- Arg0 thing affecting
- Arg1 thing affected
- Arg2 instrument

The predicate occurs 149 times in the corpus. The algorithm found Arg0 and Arg2, as well as Arg1 and Arg2, to be in complementary distribution.

Manual investigation revealed that in fact, Arg2 never appears in the corpus at all. Presumably, either Arg0 and Arg2 should be merged, or—more likely—Arg2 should not be an argument, but rather an adjunct.

## 3.6 Incidental findings

### 3.6.1 Mistakes uncovered in frame files

In the process of calculating the set of possible argument pairs for each predicate in the PropBank frame files, we found a roleset that erroneously had two Arg1s. The predicate in question was *proscribe.01*. The roles in the frame file were:

- Arg0 causer
- Arg1 thing proscribed
- Arg1 proscribed from

It was clear from the annotations in the example sentence that the "second" Arg1 was intended to be an Arg2: *[The First Amendment$_{Arg0}$] proscribes [the government$_{Arg1}$] from [passing laws abridging the right to free speech$_{\mathbf{Arg2}}$].*

### 3.6.2 Unlicensed arguments used in the corpus

We found eighteen tokens in the corpus that were annotated with argument structures that were not licensed by the roleset for the corresponding predicate. For example, the predicate *zip.01* has only a single argument in its semantic representation—Arg0, described as *entity in motion.* However, the corpus contains a token of *zip.01* that is annotated with an Arg0 and an Arg1.

# 4 Discussion/Conclusions

## 4.1 The effect of the argument/adjunct distinction

The validity and usefulness of the distinction between arguments and adjuncts is an ongoing controversy in biomedical computational lexical semantics. The BioProp project (Chou et al., 2006; Tsai et al., 2006) makes considerable use of adjuncts, essentially identically to PropBank; however, most biomedical PAS-oriented projects have relatively larger numbers of arguments and lesser use of adjuncts (Wattarujeekrit et al., 2004; Kogan et al., 2005; Shah et al., 2005) than PropBank. Overall, one would predict fewer non-coöccurring arguments with a set of representations that made a stronger distinction between arguments and adjuncts; overall arity of rolesets would be smaller (see above for the effect of arity on the number of observations required for a predicate), and the arguments for such a representation might be more "core" to the semantics of the predicate, and might therefore be less likely to not occur overall, and therefore less likely to not coöccur.

## 4.2 The effect of syntactic representation on observed argument distributions

The original work by Cohen and Hunter assumed a very simple, and very surface, syntactic representation. In particular, there was no representation of traces. In contrast, PropBank is built on Treebank II, which does include representation of traces, and arguments can, in fact, be filled by traces. This could be expected to reduce the number of tokens of apparently absent arguments, and thereby the number of non-coöoccurring arguments. This doesn't seem to have had a strong enough effect to interfere with the ability of the method to uncover errors.

## 4.3 The effect of arity

The mode for distribution of arities in the PropBank framefiles was 2 (see Table 3). In contrast, the modes for distribution of rolesets with at least one argument pair in complementary distribution across arities and for distribution of argument pairs in complementary distribution across arities was 4 or 5 for the full range of minimum observations of the predicates from 200 to 10 (data omitted for space).

This supports the initial assumption that higher-arity predicates are more likely to have argument pairs in complementary distribution—see Section 1.2 above.

One aspect of a granular analysis of the data is worth pointing out with respect to the effects of arity: as a validation check, note that for all arities, the number of predicates and the number of argument pairs rises as the minimum required number of tokens of the predicate in the corpus goes down.

### 4.4 Conclusions

The goals of this study were to investigate the automatability and scalability of a technique for PAS quality assurance that had previously only been shown to work for a small lexical resource and a small corpus, and to use it to characterize the quality of the shallow semantic representations in the PropBank framefiles. The evaluation procedure was found to be automatable: the process of finding argument pairs in complementary distribution is achievable by running a single Java application. In addition, the use of a common representation for argument sets in a framefile and argument sets in a PropBank annotation enabled the fortuitous discovery of a number of problems in the framefiles and in the corpus (see Section 3.6) as a side-effect of application of the technique.

The process was also found to scale well, with a running time of less than one and a half minutes for a set of 4,654 rolesets and a 1-million-word corpus on a moderately priced laptop; additionally, the resource maintainer's efforts can easily be focussed towards the most likely and the most prevalent error sources by adjusting the minimum number of observations required before reporting a case of complementary distribution. The process was also found to be able to identify missing sense distinctions and to identify bad arguments.

In addition to our findings regarding the quality assurance technique, a granular breakdown of the errors found by the algorithm by arity and minimum number of observations (data not shown due to space) allows us to estimate the number of errors in the PropBank framefiles. A reasonable upper-bound estimate for the number of errorful rolesets is the number of predicates that were observed at least 10 times and were found to have at least one pair of arguments in complementary distribution (the bottom row of Table 4), adjusted by the accuracy of the technique that we reported in Section 3.1, i.e. 0.69. This yields a worst-case scenario of (0.69*328)/4,654 rolesets, or 4.9% of the rolesets in PropBank, being in need of revision. The best-case scenario would assume that we can only draw conclusions about the predicates with high numbers of observations and high arity, again adjusted downward for the accuracy of the technique; taking 5 or more arguments as high arity, this yields a best-case scenario of (0.69*17)/4,654 rolesets, or 0.3% of the rolesets in PropBank, being in need of revision. A different sort of worst-case scenario assumes that the major problem in maintaining a proposition bank is not fixing inadequate representations, but *finding* them. On this assumption, the problematic representations are the ones with small numbers of tokens and low arity. Taking 3 or fewer arguments as low arity yields a worst-case scenario of 99/4,654 rolesets (no adjustment for accuracy required), or 2.13% of the rolesets in PropBank, being essentially uncharacterizable as to the goodness of their semantic representation[1].

Besides its obvious role in quality assurance for proposition banks, there may be other uses for this technique, as well. The output of the technique may also be useful in sense grouping and splitting and in detecting metaphorical uses of verbs (e.g. the *accept* example). As the PropBank model is extended to an increasingly large set of languages (currently Arabic, Basque, Catalan, Chinese, Hindi, Korean, and Russian), the need for a quality assurance mechanism for proposition banks—both to ensure the quality of their contents, and to assure funding agencies that they are evaluatable—will only grow larger.

## References

Ron Artstein and Massimo Poesio. 2008. Inter-coder agreement for computational linguistics. *Computational Linguistics*, 34(4):555–596.

Jane Austen. 1811. *Sense and Sensibility*.

Xavier Carreras and Lluís Màrquez. 2005. Introduction to the CoNLL-2005 shared task: semantic role label-

[1] The situation is arguably actually somewhat worse than this, since it does not take into account predicates which occur fewer than ten times in the corpus; however, there is a reasonable counter-argument that those predicates are too rare for any individual roleset to have a large impact on the overall goodness of the resource.

ing. In *Proceedings of the 9th conference on computational natural language learning*, pages 152–164.

Werner Ceusters, Barry Smith, Anand Kumar, and Christoffel Dhaen. 2004. Mistakes in medical ontologies: where do they come from and how can they be detected? In D.M. Pisanelli, editor, *Ontologies in medicine: proceedings of the workshop on medical ontologies*, pages 145–163. IOS Press.

Wen-Chi Chou, Richard Tzong-Han Tsai, Ying-Shan Su, Wei Ku, Ting-Yi Sung, and Wen-Lian Hsu. 2006. A semi-automatic method for annotating a biomedical proposition bank. In *Proceedings of the workshop on frontiers in linguistically annotated corpora 2006*, pages 5–12. Association for Computational Linguistics.

J.J. Cimino, H. Min, and Y. Perl. 2003. Consistency across the hierarchies of the UMLS Semantic Network and Metathesaurus. *Journal of Biomedical Informatics*, 36:450–461.

James J. Cimino. 1998. Auditing the Unified Medical Language System with semantic methods. *Journal of the American Medical Informatics Association*, 5:41–51.

James J. Cimino. 2001. Battling Scylla and Charybdis: the search for redundancy and ambiguity in the 2001 UMLS Metathesaurus. In *Proc. AMIA annual symposium*, pages 120–124.

K. Bretonnel Cohen and Lawrence Hunter. 2006. A critical revew of PASBio's argument structures for biomedical verbs. *BMC Bioinformatics*, 7(Suppl. 3).

K. B. Cohen, Lynne Fox, Philip V. Ogren, and Lawrence Hunter. 2005a. Corpus design for biomedical natural language processing. In *Proceedings of the ACL-ISMB workshop on linking biological literature, ontologies and databases*, pages 38–45. Association for Computational Linguistics.

K. Bretonnel Cohen, Lynne Fox, Philip V. Ogren, and Lawrence Hunter. 2005b. Empirical data on corpus design and usage in biomedical natural language processing. In *AMIA 2005 symposium proceedings*, pages 156–160.

George Hripcsak and Adam S. Rothschild. 2005. Agreement, the F-measure, and reliability in information retrieval. *Journal of the American Medical Informatics Association*, 12(3):296–298.

Nancy Ide and Chris Brew. 2000. Requirements, tools, and architectures for annotated corpora. In *Proc. data architectures and software support for large corpora*, pages 1–5.

Paul Kingsbury and Martha Palmer. 2002. From TreeBank to PropBank. In *Proceedings of the LREC*.

Paul Kingsbury, Martha Palmer, and Mitch Marcus. 2002. Adding semantic annotation to the Penn TreeBank. In *Proceedings of the Human Language Technology Conference*.

Richard Kittredge. 1982. Variation and homogeneity of sublanguages. In Richard Kittredge and John Lehrberger, editors, *Sublanguage: studies of language in restricted semantic domains*, pages 107–137.

Yacov Kogan, Nigel Collier, Serguei Pakhomov, and Michael Krauthammer. 2005. Towards semantic role labeling & IE in the medical literature. In *AMIA 2005 Symposium Proceedings*, pages 410–414.

Jacob Kohler, Katherine Munn, Alexander Ruegg, Andre Skusa, and Barry Smith. 2006. Quality control for terms and definitions in ontologies and taxonomies. *BMC Bioinformatics*, 7(1).

Maria Koptjevskaja-Tamm. 1993. *Nominalizations*. Routledge.

Geoffrey Leech. 1993. Corpus annotation schemes. *Literary and linguistic computing*, pages 275–281.

Mitchell P. Marcus, Mary A. Marcinkiewicz, and Beatrice Santorini. 1993. Building a large annotated corpus of English: the Penn Treebank. *Computational Linguistics*, 19(2):313–330.

Adam Meyers, Ruth Reeves, Catherine Macleod, Rachel Szekely, Veronika Zielinska, Brian Young, and Ralph Grishman. 2004. Annotating noun argument structure for NomBank. In *Proceedings of Language Resources and Evaluation, LREC*.

Philip V. Ogren, K. Bretonnel Cohen, George K. Acquaah-Mensah, Jens Eberlein, and Lawrence Hunter. 2004. The compositional structure of Gene Ontology terms. *Pacific Symposium on Biocomputing*, pages 214–225.

Martha Palmer, Paul Kingsbury, and Daniel Gildea. 2005. The Proposition Bank: an annotated corpus of semantic roles. *Computational Linguistics*, 31(1):71–106.

Martha Palmer, Stephanie Strassel, and Randee Tangi. Download date December 17, 2010. Historical development and future directions in data resource development. In *MINDS 2006–2007*.

Parantu K. Shah, Lars J. Jensen, Stéphanie Boué, and Peer Bork. 2005. Extraction of transcript diversity from scientific literature. *PLoS Computational Biology*, 1(1):67–73.

Richard Tzong-Han Tsai, Wen-Chi Chou, Yu-Chun Lin, Cheng-Lung Sung, Wei Ku, Ying-Shan Su, Ting-Yi Sung, and Wen-Lian Hsu. 2006. BIOSMILE: adapting semantic role labeling for biomedical verbs: an exponential model coupled with automatically generated template features. In *Proceedings of the BioNLP Workshop on Linking Natural Language Processing and Biology*, pages 57–64. Association for Computational Linguistics.

Tuangthong Wattarujeekrit, Parantu K. Shah, and Nigel Collier. 2004. PASBio: predicate-argument structures for event extraction in molecular biology. *BMC Bioinformatics*, 5(155).

Martin Wynne, editor. 2005. *Developing linguistic corpora: a guide to good practice*. David Brown Book Company.