Combining Lexical Resources: Mapping Between PropBank and VerbNet

Edward Loper, Szu-ting Yi, and Martha Palmer

Abstract

A wide variety of lexical resources have been created to allow automatic semantic processing of novel text. However, each resource has its own practical and theoretical idiosyncracies, making it difficult to combine the information from different resources. We discuss the form that these differences can take, and describe how we overcame some of them in creating a mapping between two important resources: PropBank and VerbNet. Furthermore, we present experimental results that show that this mapping improves performance for PropBank-style semantic role labeling. Since PropBank was designed on a verb-by-verb basis, the argument labels Arg2 - Arg5 get used for a wide variety of argument roles. As a result, it can be difficult for automatic classifiers to learn to distinguish these arguments. But by using the mapping that we have created between PropBank and VerbNet, we can train a classifier based on VerbNet argument labels, which are more consistent and therefore easier to learn.

1 Introduction

A wide variety of computational semantics tasks depend on information about specific lexical items, and their relationships to other lexical items. For example, in order to determine that sentence (1b) answers question (1a), we must know that the verb "located" is related to the verb "covered;" and we must know how their arguments are related.

(1) a. Where are the grape arbors located?

b. Every path from back door to yard was covered by a grape-arbor, and every yard had fruit trees.

In order to capture this kind of lexeme-specific information, several lexical resources have been manually created. The projects that created these resources are driven by different goals, are applied to different types of data (including different genres), and are created by people with different intellectual backgrounds. One result of this diversity is that each resource contains information about lexemes that is not present in the other resources. By combining the information from multiple resources, we can learn more about individual lexical items, and more effectively use that information for concrete tasks such as question answering.

Another result of the diversity in the projects' backgrounds is that different projects tend to make different underlying theoretical assumptions; and sometimes these assumptions are not directly compatible. An important example of this theoretical disagreement is what level of word sense granularity should be used to define senses of lexemes. Typically, any two resources will use different criteria to divide words into word senses; and as a result, there is often no direct one-to-one mapping between the senses of the lexemes of any two corpora.

In this paper, we discuss how these difficulties can be overcome, and describe a mapping that we have created between two important lexical resources, PropBank and VerbNet. We show how this mapping can be used to increase performance on the task of semantic role labeling.

2 Lexical Resources

2.1 PropBank

PropBank [9] is an annotation of one million words of the Wall Street Journal portion of the Penn Treebank II [7] with predicate-argument structures for verbs, using semantic role labels for each verb argument. In order to remain theory neutral, and to increase annotation speed, role labels were defined on a per-lexeme basis. Although the same tags were used for all verbs, (namely Arg0, Arg1, ..., Arg5), these tags are meant to have a verb-specific meaning.

Thus, the use of a given argument label should be consistent across different uses of that verb, including syntactic alternations. For example, the Arg1 (underlined) in "John broke the window" is the same window that is annotated as the Arg1 in "The window broke", even though it is the syntactic subject in one sentence and the syntactic object in the other.

There is no guarantee that an argument label will be used consistently across different verbs. For example, the Arg2 label is used to designate the *destination* of the verb "bring;" but the *extent* of the verb "rise." Generally, the arguments are simply listed in the order of their prominence for each verb. However, an explicit effort was made when PropBank was created to use Arg0 for arguments that fulfill Dowty's criteria for "prototypical agent," and Arg1 for arguments that fulfill the criteria for "prototypical patient." [3] As a result, these two argument labels are significantly more consistent across verbs than the other three. But nevertheless, there are still some inter-verb inconsistencies for even Arg0 and Arg1.

PropBank divides words into lexemes using a very coarse-grained sense disambiguation scheme: two senses are only considered different if their argument labels are different. For example, PropBank distinguishes the "render inoperable" sense of "break" from the "cause to fragment" sense. In PropBank, each word sense is known as a "frameset." Information about each frame, including descriptions of the verb-specific meaning for each argument tag (Arg0, . . . , Arg5), is defined in "frame files" that are distributed with the corpus.

PropBank's model of predicate argument structures differs from dependency parsing in that verbs are treated independently. In dependency parsing, each phrase can be dependent on only one other phrase; but since PropBank describes each verb in the sentence independently, a single argument may be used for multiple predicates. For example, in the following sentence, PropBank would use the phrase "his dog" as the argument to two predicates, "scouted" and "chasing;" but in traditional dependency parsing models, this would not be allowed, since each phrase can be dependant on only one other phrase.

(2) a. His dog **scouted** ahead, chasing its own mangy shadow.

b. His dog scouted ahead, **chasing** its own mangy shadow.

The primary goal of PropBank is to provide consistent, general purpose labeling of semantic roles for a large quantity of coherent text that can provide training data for supervised machine learning algorithms, in the same way the Penn Treebank has supported the training of statistical syntactic parsers. PropBank can provide frequency counts for (statistical) analysis or generation components for natural language applications. In addition to the annotated corpus, PropBank provides a lexicon which divides each word into corse-grained senses known as "framesets," describes the argument roles that can be used with each frameset, and provides example usages in a variety of syntactic contexts. This lexical resource is used as a set of verb-specific guidelines by the annotators, and can be seen as quite similar in nature to FrameNet, although much more coarse-grained and general purpose in the specifics.

2.2 VerbNet

VerbNet [12] consists of hierarchically arranged verb classes, inspired by and extended from classes of Levin 1993 [6]. Each class and subclass is characterized extensionally by its set of verbs, and intensionally by a list of the arguments of those verbs and syntactic and semantic information about the verbs. The argument list consists of thematic roles (23 in total) and possible selectional restrictions on the arguments expressed using binary predicates. The syntactic information maps the list of thematic arguments to deep-syntactic arguments (i.e., normalized for voice alternations, and transformations). The semantic predicates describe the participants during various stages of the event described by the syntactic frame.

The same thematic role can occur in different classes, where it will appear in different predicates, providing a class-specific interpretation of the role. VerbNet has been extended from the original Levin classes, and now covers 4526 senses for 3769 lexemes. A primary emphasis for VerbNet is the grouping of verbs into classes that have a coherent syntactic and semantic characterization, that will eventually facilitate the acquisition of new class members based on observable syntactic and semantic behavior. The hierarchical structure and small number of thematic roles is aimed at supporting generalizations.

3 Mapping PropBank to VerbNet

Because PropBank includes a large corpus of manually annotated predicate-argument data, it can be used to train supervised machine learning algorithms, which can in turn provide PropBank-style annotations for novel or unseen text. However, PropBank lacks much of the information that is contained in VerbNet, including information about selectional restrictions, verb semantics, and inter-verb relationships. We have therefore created a mapping between VerbNet and PropBank, which will allow us to use the machine learning techniques that have been developed for PropBank annotations to generate more semantically abstract VerbNet representations.

The mapping between VerbNet and PropBank consists of two parts: a *lexical mapping* and an *instance classifier*. The lexical mapping is responsible for specifying the potential mappings between PropBank and VerbNet for a given word; but it does not specify which of those mappings should be used for any given occurrence of the word. That is the job of the instance classifier, which looks at the word in context, and decides which of the mappings is most appropriate. In essence, the instance classifier is performing word sense disambiguation, deciding which lexeme from each database is correct for a given occurrence of a word.

3.1 The Lexical Mapping

The lexical mapping defines the set of *possible* mappings between the two lexicons for a given word, independent of context. For each lexeme in PropBank, it provides a list of the VerbNet lexemes that can be used to describe word sense covered by that PropBank lexeme. For example, the first sense of “take” in PropBank (“take, acquire, come to have”) can map to either VerbNet class 10.5 (“steal”) or VerbNet class 11.3 (“bring”).

For each pairing between a PropBank lexeme and a VerbNet lexeme, the lexical mapping further specifies how the argument roles are mapped. For example, it provides the following role mapping between PropBank’s first sense of “take” and VerbNet’s “bring-11.3” sense:

(3)

PropBank Role	VerbNet Role
Arg0	Agent
Arg1	Theme
Arg2	Source

Although the lexical mapping is expressed as a mapping from PropBank to VerbNet, it can also be used in reverse. In particular, it can be used to find a list of the PropBank lexemes that correspond to a given VerbNet lexeme; and to find the argument role mappings for each sense pair. Thus, the lexical mapping is actually bi-directional.

Unfortunately, there are a number of issues that can prevent us from generating a complete mapping between PropBank and VerbNet for any given word. One important issue is differences in the coverage of the two resources. Each resource contains many words that are not described at all in the other resource; and even if both resources describe a word, they may describe different senses of it. For example, the PropBank entry for “barge” describes the sense used in sentence (4a); but the VerbNet entry describes the somewhat less common sense used in sentence (4b).

(4) a. John barged into the room.

b. John barged the lumber across the river.

If a given word sense is not covered by both PropBank and VerbNet, then it is impossible to define a mapping for that sense. At the time the mapping was created, 25.5% of the word instances in PropBank were not covered by VerbNet. However, work is underway to extend the coverage of both VerbNet and PropBank, which will fill in these gaps and allow for a more complete mapping.

Another important issue is mismatches between the arguments in PropBank and VerbNet. These mismatches can take one of three forms: an argument described by one resource may be omitted by the other; a single argument described by one resource may be split into multiple arguments by the other; or the two resources may disagree entirely on how arguments should be defined. Luckily, this last case appears to be quite rare. However, there are numerous examples of omitted and split arguments.

Omitted arguments often reflect simple oversight – an argument may not have been seen or considered when creating the resource. However, some omissions reflect theoretical differences between PropBank and VerbNet. This can have one of two causes: first, PropBank and VerbNet sometimes differ on where they draw the line between "arguments," which are listed with each lexeme, and "adjuncts," which are not; and second, there are cases where one resource's lexeme reflects a more specific sense than the one it is mapped to, and a given argument is not applicable within that specific sense.

Split arguments typically reflect theoretical differences between the two resources: one resource considers two arguments to be "the same," while the other resource considers them to be different. In these cases, it can be productive to re-consider the decisions made by each resource, to see if either one should be changed to match the other. Often linguistic tests can be used to decide which argument set is more plausible. For example, if the split arguments are mutually exclusive (i.e., only one can appear in any given sentence), then we might give preference to the interpretation that they are really all one argument.

3.2 Instance Classifier

The instance classifier is responsible for deciding which of the possible mappings that are defined by the lexical mapping is applicable to a given occurrence of a word. Instead of directly providing an instance classifier, we have decided to annotate training data that can be used with supervised machine learning methods to create the instance classifier. This should allow for experimentation with what features and learning methods are most useful for instance classification.

The training data for the instance classifier consists of parallel PropBank and VerbNet class annotations for all verbs in the Wall Street portion of the Penn Treebank II.[1] Since this data is already included in PropBank, only the VerbNet class labels needed to be annotated. We added these class labels using a semi-automatic process, where two heuristic classifiers generated an initial labelling, and human annotators hand-corrected their output.

The first of these heuristic classifiers works by running the SenseLearner WSD engine to find the WordNet class of each verb, and then using the existing WordNet/VerbNet mapping to choose the corresponding VerbNet class. This heuristic is limited by the performance of the WSD engine, and by the fact that the WordNet/VerbNet mapping is not available for all VerbNet verbs. The second heuristic classifier works by examining the syntactic context for each verb instance, and comparing it to the syntactic frames defined by each VerbNet class. The VerbNet class with a syntactic frame that most closely matches the instance's context is assigned to the instance. Having defined these two heuristic methods, we ran them on the Penn Treebank corpus. We then hand-corrected the results, in order to obtain a VerbNet-annotated version of the Treebank corpus.

4 Using the Mapping to Achieve Robust Semantic Role Labeling

4.1 Semantic Role Labeling

Correctly identifying semantic entities and successfully disambiguating the relations between them and their predicates is an important and necessary step for successful natural language processing applications, such as text summarization, question answering, and machine translation. An important part of this task is *Semantic Role Labeling* (SRL), where the goal is to locate the constituents which are arguments of a given verb, and to assign them appropriate semantic roles that describe how they relate to the verb. Many researchers have investigated using machine learning for this task since 2000 [2, 4, 5, 8, 14, 10, 11, 13]. For two years, the CoNLL workshop has made this problem the shared task [1].

4.2 Current Issues of SRL

Current SRL system performance on the Wall Street Journal corpus seems to be reaching a ceiling. However, the Wall Street Journal (WSJ) corpus

[1] Excepting verbs whose senses are not present in VerbNet (24.5% of instances).

is highly specialized, and tends to use genre-specific word senses for many verbs. As a result, the SRL systems that are trained on the Wall Street Journal are lacking in robustness – although they perform well on the WSJ corpus, their performance drops significantly when run on texts taken from other genres. The 2005 CoNLL shared task has addressed this issue of robustness by evaluating participating systems on a test set extracted from the Brown corpus, which is very different from the WSJ corpus that was used for training. The results suggest that there is much work to be done in order to improve system robustness.

One of the reasons that current SRL systems have difficulty deciding which role label to assign to a given argument is that role labels are defined on a per-verb basis. This is less problematic for Arg0 and Arg1, where a conscious effort was made to be consistent across verbs; but is a significant problem for Args[2-5], which tend to have very verb-specific meanings. This problem is exacerbated even further on novel genres, where SRL systems are more likely to encounter uses of arguments that were unseen in the training data.

4.3 Addressing Current SRL Problems via Lexical Mappings

By exploiting the mapping between PropBank and VerbNet, we can transform the data to make it more consistent, and to expand the size and variety of the training data. In particular, we can use the mapping to transform the verb-specific PropBank role labels into the more general thematic role labels that are used by VerbNet. Unlike the PropBank labels, the VerbNet labels are defined consistently across verbs; and therefore it should be easier for statistical SRL systems to model them. Furthermore, since the VerbNet role labels are significantly less verb-specific than the PropBank roles, the SRL's models should generalize better to novel verbs, and to novel uses of known verbs.

4.4 SRL Experiments on Linked Lexical Resources

In order to test these ideas, we re-trained our Maximum Entropy SRL system [14] on a transformed version of the PropBank data, where PropBank role labels were mapped to the corresponding VerbNet thematic role labels. Our hypothesis is that the mapping between VerbNet and PropBank creates more consistent training data; therefore it should improve the SRL system performance. In addition, because VerbNet thematic roles behave more consistently than PropBank argument labels across verbs, an SRL system

Group1	Group2	Group3	Group4	Group5	Group6
Theme Theme1 Theme2 Predicate Stimulus Attribute	Topic	Patient Product Patient1 Patient2	Agent Actor2 Experiencer Cause	Source Location Destination Recipient Beneficiary Material	Asset

Figure 1: Thematic Role Groupings for Arg1

Group1	Group2	Group3	Group4	Group5
Recipient Destination Location Source Material Beneficiary	Extent Asset	Predicate Attribute Theme Theme1 Theme2 Topic	Patient2 Product	Instrument Actor2 Experiencer Cause

Figure 2: Thematic Role Groupings for Arg2

trained on VerbNet thematic roles should be better able to generalize to new instances, making it more robust on genres other than WSJ-style articles.

We conducted two sets of experiments: one to test the effect of the mapping on learning Arg2; and one to test the effect on learning Arg1. Since Arg2 is used in very verb-dependent ways, we expect that mapping it to VerbNet role labels will increase performance. However, since a conscious effort was made to keep the meaning of Arg1 consistent across verbs, we should expect that mapping it to VerbNet labels will result in a smaller improvement.

Each experiment compares two SRL systems: one trained using the original PropBank role labels; the other trained with the argument role under consideration (Arg1 or Arg2) subdivided based on which VerbNet role label it maps to. In order to prevent the training data from these subdivided labels from becoming too sparse (which would impair system performance) we grouped similar thematic roles together. [2] The argument role groupings we used are shown in Figure 1 and Figure 2.

The training data for both experiments is the portion of Penn Treebank II (sections 02-21) that is covered by the mapping. We evaluated each experimental system using two test sets: section 23 of the Penn Treebank II, which represents the same genre as the training data; and the PropBanked portion of the Brown corpus, which represents a very different genre.

[2] Karin Kipper assisted in creating the groupings.

5 Results and Discussion

Table 1 describes the results of SRL overall performance tested on the WSJ corpus Section 23; Table 2 demonstrates the SRL overall system performance tested on the Brown corpus. Systems Arg1-Original and Arg2-Original are trained using the original PropBank labels, and show the baseline performance of our SRL system. Systems Arg1-Mapped and Arg2-Mapped are trained using PropBank labels augmented with VerbNet thematic role groups. In order to allow comparison between the system using original PropBank labels and the systems that augmented those labels with VerbNet theta role groups, system performance was evaluated based solely on the PropBank role label that was assigned.

System	**Precision**	**Recall**	**F1**
Arg1-Original	89.24	77.32	82.85
Arg1-Mapped	90.00	76.35	82.61
Arg2-Original	73.04	57.44	64.31
Arg2-Mapped	84.11	60.55	70.41

Table 1: SRL System Performance on Arg1 Mapping and Arg2 Mapping, tested using the *WSJ corpus (section 23)*. This represents performance on the same genre as the training corpus.

System	**Precision**	**Recall**	**F1**
Arg1-Original	86.01	71.46	78.07
Arg1-Mapped	88.24	71.15	78.78
Arg2-Original	66.74	52.22	58.59
Arg2-Mapped	81.45	58.45	68.06

Table 2: SRL System Performance on Arg1 Mapping and Arg2 Mapping, tested using the *PropBank-ed Brown corpus*. This represents performance on a different genre from the training corpus.

We had hypothesized that with the use of thematic roles, we would be able to create a more consistent training data set which would result in an improvement in system performance. In addition, the thematic roles would behave more consistently than overloaded Args[2-5] across verbs, which should enhance robustness. However, since in practice we are also increasing the number of argument labels an SRL system needs to tag, the system might suffer from data sparseness. Our hope is that the enhancement gained from the mapping will outweigh the loss due to data sparseness.

From Table 1 and Table 2 we see the F1 scores of Arg1-Original and Arg1-Mapped are statistically indifferent both on the WSJ corpus and the Brown corpus. These results confirm the observation that Arg1 in the PropBank behaves fairly verb-independently so that the VerbNet mapping does not provide much benefit. The increase of precision due to a more coherent training data set is compensated for by the loss of recall due to data sparseness.

The results of the Arg2 experiments tell a different story. Both precision and recall are improved significantly, which demonstrates that the Arg2 label in the PropBank is quite overloaded. The Arg2 mapping improves the overall results (F1) on the WSJ by 6% and on the Brown corpus by almost 10%. As a more diverse corpus, the Brown corpus provides many more opportunities for generalizing to new usages. Our new SRL system handles these cases more robustly, demonstrating the consistency and usefulness of the thematic role categories.

References

[1] Xavier Carreras and Lluís Márquez. Introduction to the conll-2005 shared task: Semantic role labeling. In *Proceedings of CoNLL*, 2005.

[2] John Chen and Owen Rambow. Use of deep linguistic features for the recognition and labeling of semantic arguments. In *Proceedings of EMNLP-2003*, Sapporo, Japan, 2003.

[3] D. R. Dowty. Thematic proto-roles and argument selection. *Language*, 67:574–619, 1991.

[4] Daniel Gildea and Julia Hockenmaier. Identifying semantic roles using Combinatory Categorial Grammar. In *2003 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 57–64, Sapporo, Japan, 2003.

[5] Kadri Hacioglu, Sameer Pradhan, Wayne Ward, James H. Martin, and Daniel Jurafsky. Shallow semantic parsing using support vector machines. Technical report, The Center for Spoken Language Research at the University of Colorado (CSLR), 2003.

[6] Beth Levin. *English Verb Classes and Alternations: A Preliminary Investigation*. The University of Chicago Press, 1993.

[7] M. Marcus, G. Kim, M. Marcinkiewicz, R. MacIntyre, A. Bies, M. Ferguson, K. Katz, and B. Schasberger. The Penn treebank: Annotating predicate argument structure, 1994.

[8] Alessandro Moschitti. A study on convolution kernel for shallow semantic parsing. In *Proceedings of the 42-th Conference on Association for Computational Linguistic (ACL-2004)*, Barcelona, Spain, 2004.

[9] Martha Palmer, Daniel Gildea, and Paul Kingsbury. The proposition bank: A corpus annotated with semantic roles. *Computational Linguistics*, 31(1):71–106, 2005.

[10] S. Pradhan, W. Ward, K. Hacioglu, J. Martin, and D. Jurafsky. Semantic role labeling using different syntactic views. In *Proceedings of the Association for Computational Linguistics 43rd annual meeting (ACL-2005)*, Ann Arbor, MI, 2005.

[11] V. Punyakanok, D. Roth, and W. Yih. The necessity of syntactic parsing for semantic role labeling. In *Proceedings of the 19th International Joint Conference on Artificial Intelligence (IJCAI-05)*, 2005.

[12] Karin Kipper Schuler. *VerbNet: A broad-coverage, comprehensive verb lexicon.* PhD thesis, University of Pennsylvania, 2005.

[13] Kristina Toutanova, Aria Haghighi, and Christopher D. Manning. Joint learning improves semantic role labeling. In *Proceedings of the Association for Computational Linguistics 43rd annual meeting (ACL-2005)*, Ann Arbor, MI, 2005.

[14] Szu-ting Yi and Martha Palmer. Pushing the boundaries of semantic role labeling with svm. In *Proceedings of the International Conference on Natural Language Processing*, 2004.