Mining of Parsed Data to Derive Deverbal Argument Structure

Olga Gurevich **Scott A. Waterman**
Microsoft / Powerset
475 Brannan Street, Ste. 330
San Francisco, CA 94107
{olya.gurevich,scott.waterman}@microsoft.com

Abstract

The availability of large parsed corpora and improved computing resources now make it possible to extract vast amounts of lexical data. We describe the process of extracting structured data and several methods of deriving argument structure mappings for deverbal nouns that significantly improves upon non-lexicalized rule-based methods. For a typical model, the F-measure of performance improves from a baseline of about 0.72 to 0.81.

1 Introduction

There is a long-standing division in natural language processing between symbolic, rule-based approaches and data-driven, statistical ones. Rule-based, human-curated approaches are thought to be more accurate for linguistic constructions explicitly covered by the rules. However, such approaches often have trouble scaling up to a wider range of phenomena or different genres of text. There have been repeated moves towards hybridized approaches, in which rules created with human linguistic intuitions are supplemented by automatically derived corpus data (cf. (Klavans and Resnik, 1996)).

Unstructured corpus data for English can easily be found on the Internet. Large corpora of text annotated with part of speech information are also available (such as the British National Corpus). However, it is much harder to find widely available, large corpora annotated for syntactic or semantic structure. The Penn Treebank (Marcus et al., 1993) has until recently been the only such corpus, covering 4.5M words in a single genre of financial reporting. At the same time, the accuracy and speed of syntactic parsers has been improving greatly, so that in recent years it has become possible to automatically create parsed corpora of reasonable quality, using much larger amounts of text with greater genre variation. For many NLP tasks, having more training data greatly improves the quality of the resulting models (Banko and Brill, 2001), even if the training data are not perfect.

We have access to the entire English-language text of Wikipedia (about 2M pages) that was parsed using the XLE parser (Riezler et al., 2002), as well as an architecture for distributed data-mining within this corpus, called Oceanography (Waterman, 2009). Using the parsed corpus, we extract a large volume of dependency relations and derive lexical models that significantly improve a rule-based system for determining the underlying argument structure of deverbal noun constructions.

2 Deverbal Argument Mapping

Deverbal nouns, or nominalizations, are nouns that designate some aspect of the event referred to by the verb from which they are morphologically derived (Quirk et al., 1985). For example, the noun *destruction* refers to the action described by the verb *destroy*, and *destroyer* may refer to the agent of that event. Deverbal nouns are very common in English texts: by one count, about half of all sentences in written text contain at least one deverbal noun (Gurevich et al., 2008). Thus, a computational system that aims to match multiple ways of expressing the same underlying events (such as question answering or search) must be able to deal with deverbal nouns.

To interpret deverbal constructions, one must be able to map nominal and prepositional modifiers to the various roles in the verbal frame. For intransitive verbs, almost any argument of the deverbal noun is mapped to the verb's subject, e.g. *abundance of food* gives rise to *subj(abound, food)*. If the underlying verb is transitive, and the deverbal noun has two arguments, the mappings are also fairly straightforward. For example, the phrase *Carthage's defeat by Rome* gives rise to

Proceedings of the 2009 Workshop on Grammar Engineering Across Frameworks, ACL-IJCNLP 2009, pages 19–27,
Suntec, Singapore, 6 August 2009. ©2009 ACL and AFNLP

the arguments *subj(defeat, Rome)* and *obj(defeat, Carthage)*, based on knowledge that a “by” argument usually maps to the subject, and the possessive in the presence of a “by” argument usually maps to the object (Nunes, 1993).

However, in many cases a deverbal noun has only one argument, even though the underlying verb may be transitive. In such cases, our system has to decide whether to map the lone argument of the deverbal onto the subject or object of the verb. This mapping is in many cases obvious to a human: e.g., *the king’s abdication* corresponds to *subj(abdicate, king)*, whereas *the room’s adornment* corresponds to *obj(adorn, room)*. In some cases, the mapping is truly ambiguous, e.g., *They enjoyed the support of the Queen* vs. *They jumped to the support of the Queen*. Yet in other cases, the lone argument of the deverbal noun is neither the subject nor the object of the underlying verb, but it may correspond to a different (e.g. prepositional) argument of the verb, as in *the travels of 1996* (corresponding to *someone traveled in 1996*). Finally, in some cases the deverbal noun is being used in a truly nominal sense, without an underlying mapping to a verb, as in *Bill Gates’ foundation*, and the possessive is not a verbal argument.

The predictive models in this paper focus on this case of single arguments of deverbal nouns with transitive underlying verbs. To constrain the scope of the task, we focus on possessive arguments, like *the room’s adornment*, and ‘of’ arguments, like *the support of the Queen*. Our goal is to improve the accuracy of verbal roles assigned in such cases by creating lexically-specific preferences for individual deverbal noun / verb pairs. Some of our experiments also take into account some lexical properties of the deverbal noun’s arguments. The lexical preferences are derived by comparing argument preferences of verbs with those of related deverbal nouns, derived from a large parsed corpus using Oceanography.

2.1 Current Deverbal Mapping System

We have a list of approximately 4000 deverbal noun / verb pairs, constructed from a combination of WordNet’s derivational links (Fellbaum, 1998), NomLex (Macleod et al., 1998), NomLexPlus (Meyers et al., 2004b) and some independent curation. In the current system implementation, we attempt to map deverbal nouns onto corresponding verbs using a small set of heuristics described in (Gurevich et al., 2008). We distinguish between event nouns like *destruction*, agentive nouns like *destroyer*, and patient-like nouns like *employee*.

If a deverbal noun maps onto a transitive verb and has only one argument, the heuristics are as follows. Arguments of agentive nouns become objects while the nouns themselves become subjects, so *the ship’s destroyer* maps to *subj(destroy, destroyer); obj(destroy, ship)*. Arguments of patient-like nouns become subjects while the nouns themselves become objects, so *the company’s employee* becomes *subj(employ, company); obj(employ, employee)*.

The difficult case of event nouns is currently handled through default mappings: possessive arguments become subjects (e.g., *his confession* $\mapsto$ *subj(confess, he)*), and ‘of’ arguments become objects (e.g., *confession of sin* $\mapsto$ *obj(confess, sin)*). However, as we have seen from examples above, these defaults are not always correct. The correct mapping depends on the lexical nature of the deverbal noun and its corresponding verb, and possibly on properties of the possessive or ‘of’ argument as well.

2.2 System Background

The deverbal argument mapping occurs in the context of a larger semantic search application, where the goal is to match alternate forms expressing similar concepts. We are currently processing the entire text of the English-language Wikipedia, consisting of about 2M unique pages.

Parsing in this system is done using the XLE parser (Kaplan and Maxwell, 1995) and a broad-coverage grammar of English (Riezler et al., 2002; Crouch et al., 2009), which produces constituent structures and functional structures in accordance with the theory of Lexical-Functional Grammar (Dalrymple, 2001).

Parsing is followed by a semantic processing phase, producing a more abstract argument structure. Semantic representations are created using the *Transfer* system of successive rewrite rules (Crouch and King, 2006). Numerous constructions are normalized and rewritten (e.g., passives, relative clauses, etc.) to maximize matching between alternate surface forms. This is the step in which deverbal argument mapping occurs.

2.3 Evaluation Data

To evaluate the performance of the current and experimental argument mappings, we extracted a random set of 1000 sentences from the parsed Wikipedia corpus in which a deverbal noun had a single possessive argument. Each sentence was manually annotated with the verb role mapping between the deverbal and the possessive arguments. One of six labels were assigned:

- Subject, e.g. *John's attention*
- Object, e.g. *arrangement of flowers*
- Other: there is an underlying verb, but the relationship between the verb and the argument is neither subject nor object; these relations often appear as prepositional arguments in the verbal form, e.g. *Declaration of Delhi*
- Noun modifier: the argument modifies the nominal sense of the deverbal noun, rather than the underlying verb, although there is still an underlying event, as in *director of 25 years*
- Not deverbal: the deverbal noun is not used to designate an event in this context, e.g. *the rest of them*
- Error: the parser incorrectly identified the argument as modifying the deverbal, or as being the only argument of the deverbal

Similarly, we extracted a sample of 750 sentences in which a deverbal noun had a single 'of' argument, and annotated those manually.

The distribution of annotations is summarized in Table 1. For possessive arguments, the prevalent role was subject, and for 'of' arguments it was object.

The defaults will correctly assign the majority of arguments roles.

	Possessive	'Of'
total	1000	750
unique deverbals	423	338
subj	**511 (51%)**	158 (21%)
obj	335 (34%)	**411 (55%)**
other	28 (3%)	50 (7%)
noun mod	23 (2%)	18 (2%)
not deverbal	21 (2%)	40 (5%)
error	82 (8%)	73 (10%)

Table 1: Evaluation Role Judgements, with defaults in **bold**

2.4 Lexicalizing Role Mappings

Our basic premise is that knowledge about role-mapping behavior of particular verbs will inform the role-mapping behavior of their corresponding deverbal nouns. For example, if a particular argument of a given verb surfaces as the verb's subject more often than as object, we might also prefer the subject role when the same argument occurs as a modifier of the corresponding deverbal noun. However, as nominal modification constructions impose their own role-mapping preferences (e.g., possessives are more likely to be subjects than objects), we expect different distributions of arguments to appear in the various deverbal modification patterns. Making use of this intuition requires collecting sufficient information about corresponding arguments of verbs and deverbal nouns. This is available, given a large parsed corpus, a reasonably accurate and fast parser, and enough computing capacity. The remainder of the paper details our data extraction, model-building methods, and the results of some experiments.

3 Data Collection

Oceanography is a pattern extraction and statistics language for analyzing structural relationships in corpora parsed using XLE (Waterman, 2009). It simplifies the task of programming for NL analysis over large corpora, and the sorting, counting, and distributional analysis that often characterizes statistical NLP. This corpus processing language is accompanied by a distributed runtime, which uses cluster computing to match patterns and collect statistics simultaneously across many machines. This is implemented in a specialized distributed framework for parsing and text analysis built on top of Hadoop (D. Cutting et al.,). Oceanography programs compile down to distributed programs which run in this cluster environment, allowing the NL researcher to state declaratively the data gathering and analysis tasks.

A typical program consists of two declarative parts, a *pattern matching* specification, and a set of *statistics* declarations. The pattern matching section is written using *Transfer*, a specialized language for identifying subgraphs in the dependency structures used in XLE (Crouch and King, 2006). Transfer rules use a declarative syntax for specifying elements and their relations; in this way, it is much like a very specialized *awk* or *grep* for matching within parse trees and dependency graphs.

Statistics over these matched structures are also stated declaratively. The researcher states which sub-elements or tuples are to be counted, and the resulting compiled program will output counts. Conditional distributions and comparisons between distributions are available as well.

3.1 Training Data

Using Oceanography, we extracted two sets of relations from the parsed Wikipedia corpus, *FullWiki*, with approximately 2 million documents. A smaller 10,000-document subset, the *10K* set, was used in initial experiments. Some comparative results are shown to indicate effects of corpus size on results. Summary corpus statistics are shown in table 2. The two sets were:

1. All verb-argument pairs, using verb and argument lemmas. We recorded the verb, the argument, the kind of relation between them (e.g., subject, object, etc.), and part of speech of the argument, distinguishing also among pronouns, names, and common nouns. For each combination, we record its frequency of occurrence.

2. All deverbal-argument pairs, using deverbal noun and argument lemmas. We recorded the deverbal noun, the argument, the kind of relation (e.g., possessive, 'of', prenominal modifier, etc.) and part of speech of the argument. We record the frequency of occurrence for each combination.

Some summary statistics about the extracted data are in Table 2.

FullWiki training data	
Documents	2 million
Sentences	121,428,873
Deverbal nouns with arguments	4,596
Unique verbs with deverbals	3,280
Verbs with arguments	7,682
Deverbal - role - argument sets	21,924,405
Deverbal - argument pairs	12,773,621
Deverbals with any poss argument	3,802
Possessive deverbal - argument pairs	611,192
Most frequent: *poss(work, he)*	75,343
Deverbals with any 'of' argument	4,075
'Of' deverbal- argument pairs	2,108,082
Most frequent: *of(end, season)*	15,282
Verb - role - argument sets	72,150,246
Verb - argument pairs	40,895,810
Overlapping pairs	5,069,479
Deverbals with overlapping arguments	3,211

Table 2: Training Data

4 Assigning Roles

The present method is based on projecting argument type preferences from the verbal usage to the deverbal. The intuition is that if an argument X is preferred as the subject (object) of verb V, then it will also be preferred in the semantic frame of an occurrence (N, X) with the corresponding deverbal noun N.

We model these preferences directly using the relative frequency of subject and object occurrences of each possible argument with each verb. Even with an extremely large corpus, it is unlikely that one will find direct evidence for all such combinations, and one will need to generalize the prediction.

4.1 Deverbal-only Model

The first model, *all-arg*, specializes only for the deverbal, and generalizes over all arguments, relying on the overall preference of subject v. object for the set of arguments that appear with both verb and deverbal forms. Take as an example deverbal nouns with possessive arguments (e.g., *the city's destruction*). Given the phrase (X's N), where N is a deverbal noun related to verb V, F_d(N, V, X) is a function that assigns one of the roles $subj, obj, unknown$ to the pair (V, X). In this deverbal only model, the function depends on N and V only, and not on the argument X. F_d for a any pair (N, V) is calculated as follows:

1. Find all arguments X that occur in the construction "N's X" as well as either *subj(V, X)* or *obj(V, X)*. X, N, and V have all been lemmatized. For example, *poss(city, destruction)* occurs 10 times in the corpus; *subj(destroy, city)* occurs 3 times, and *obj(destroy, city)* occurs 12 times. This approach conflates instances of *the city's destruction*, *the cities' destruction*, *the city's destructions*, etc.

2. For each argument X, calculate the ratio between the number of occurrences of *subj(V, X)* and *obj(V, X)*. If the argument occurs as subject more than 1.5 times as often as the object, increment the count of subject-preferring arguments of N by 1. If the argument occurs as object more than 1.5 times as often as subject (as would be the case with *(destroy, city)*), increment the count of object-preferring arguments. If the ratio in frequencies of occurrence is less than the cutoff ratio

of 1.5, neither count is incremented. In addition to the number of arguments with each preference, we keep track of the total number of instances for each argument preference, summed up over all individual arguments with that preference.

3. Compare the number of subject-preferring arguments of N with the number of object-preferring arguments. If one is greater than the other by more than 1.5 times, state that the deverbal noun N has a preference for mapping its possessive arguments to the appropriate verbal role. We ignore cases where the total number of occurrences of the winning arguments is too small to be informative (in the current model, we require it to be greater than 1).

If there is insufficient evidence for a deverbal N, we fall back to the default preference across all deverbals. Subject and object co-occurrences with the verb forms are always counted, regardless of other arguments the verb may have in each sentence, on the intuition that the semantic role preference of the argument is relatively unaffected and that this will map to the deverbal construction even when the possessive is the only argument. Summary preferences for *all-args* are shown in Table 3.

The same algorithm was applied to detect argument preferences for deverbals with 'of' arguments (such as *destruction of the city*). Summary preferences are shown in Table 4.

4.2 Deverbal + Argument Animacy Model

The second model tries to capture the intuition that animate arguments often behave differently than inanimate ones: in particular, animate arguments are more often agents, encoded syntactically as subjects.

We calculated argument preferences separately for two classes of arguments: (1) animate pronouns such as *he*, *she*, *I*; and (2) nouns that were not identified as names by our name tagger. We assumed that arguments in the first group were animate, whereas arguments in the second group were not. In these experiments, we did not try to classify named entities as animate or inanimate, resulting in less training data for both classes of arguments. This strategy also incorrectly classifies common nouns that refer to people (e.g., occupation names such as *teacher*).

The results of running both models on the 10K and FullWiki training sets are in Table 3 for possessive arguments and Table 4 for 'of' arguments.

For possessives, animate arguments preferred subject role mappings much more than the average across all arguments. Inanimate arguments also on the whole preferred subject mappings, but much less strongly.

For 'of' arguments, in most cases there were more object-preferring verbs, except for verbs with animate arguments, which overwhelmingly preferred subjects. We might therefore expect there to be a difference in performance between the model that treats all arguments equally and the model that takes argument animacy into account.

Model: *all-arg*		
	10K	FullWiki
Subj-preferring	**391 (65%)**	**1786 (67%)**
Obj-preferring	207 (35%)	884 (33%)
Total	598 (100%)	2670 (100%)
Model: *animacy*		
Subj-pref animate	**370 (78%)**	**1941 (79%)**
Obj-pref animate	106 (22%)	511 (21%)
Total animate	476 (100%)	2452 (100%)
Subj-pref inanimate	45 (47%)	**990 (57%)**
Obj-pref inanimate	**51 (53%)**	748 (43%)
Total inanimate	96 (100%)	1738 (100%)

Table 3: Possessive argument preferences

Model: *all-arg*		
	10K	FullWiki
Subj-preferring	143 (30%)	839 (29%)
Obj-preferring	**328 (70%)**	**2036 (71%)**
Total	471 (100%)	2875 (100%)
Model: *animacy*		
Subj-pref animate	**70 (83%)**	**1196 (74%)**
Obj-pref animate	14 (17%)	423 (26%)
Total animate	84 (100 %)	1619 (100%)
Subj-pref inanimate	83 (23%)	699 (25%)
Obj-pref inanimate	**272 (77%)**	**2068 (75%)**
Total inanimate	355 (100%)	2767 (100%)

Table 4: 'Of' argument preferences

5 Experiments

The base system against which we compare these models uses the output of the parser, identifies deverbal nouns and their arguments, and applies the heuristics described in Section 2.1 to obtain verb roles. Recall that possessive arguments of transitive deverbals map to the subject role, and 'of' arguments map to object. Also recall that these rules apply only to eventive deverbals; mapping rules for known agentive and patient-like deverbals remain as before.

In the evaluation, the experimental models take precedence: if the model predicts an outcome, it is used. The default system behavior is used as a fallback when the model does not have sufficient evidence to make a prediction. This *stacking* of models allows the use of corpus evidence when available, and generalized defaults otherwise.

For the *animacy* model, we used our full system to detect whether the argument of a deverbal was animate (more precisely, human). In addition to the animate pronouns used to generate the model, we also considered person names, as well as common nouns that had the hypernym 'person' in WordNet. If the argument was animate and the model had a prediction, that was used. If no prediction was available for animate arguments, then the inanimate prediction was used. Failing that, the prediction falls back to the general defaults.

5.1 Possessive Arguments of Deverbal Nouns

Model predictions were compared against the hand-annotated evaluation set described in Section 2.3. For each sentence in the evaluation set, we used the models to make a two-way prediction with respect to the default mapping: is the possessive argument of the deverbal noun an underlying subject or not. We ignored test sentences marked as having erroneous parses, leaving 918 (of 1000 annotated). Since we were evaluating the accuracy of the 'subject' label, all non-subject roles (object, "other", "not a deverbal", and "nominal modifier") were in the same class. The baseline for comparison is the default 'subject' role.

The possible outcomes for each sentence were:

- True Positive: Expected role and role produced by the system are "subject"
- True Negative: Expected role is not subject, and the model did not produce the label subject. Expected role and produced role may differ (e.g. expected role may be "other", and the model may produce "object", but since neither one is "subject", this counts as correct
- False Positive: Expected role is not subject, but the model produced subject
- False Negative: Expected role is subject, but the model produced some other role

As a quick evaluation, we compared baseline and model-predicted results directly in the *surface string* of the sentences, without reparsing the sentences or using the semantic rewrite rules. The advantage of this evaluation is that it is very fast to run and is easily reproducible outside of our specialized environment. This evaluation differed from the full-pipeline evaluation in two ways: (1) it did not distinguish event deverbals from agentive and patient-like deverbals, thus possibly introducing errors, and (2) it did not look up all argument lemmas to find out their animacy. This baseline had precision of 0.56; recall of 1.0, and an F-measure of 0.72.

The complete evaluation uses our full NL pipeline, reparsing the sentences and applying all of our deverbal mapping rules as described above. The baseline for this evaluation had a precision of 0.65, recall of 0.94, and F-measure of 0.77. The differences in the two baselines are mostly due to the full-pipeline evaluation having different mapping rules for agentive and patient-like deverbals.

5.1.1 Results

Results of applying the models are summarized in Table 5, for all models, trained with both the smaller and the larger data sets, and measured with and without using the full pipeline.

All models performed better than the baseline. The *all-arg* model did about the same as the *animacy* model with both training sets. We suggest some reasons for this in the next section.

It is unambiguously clear that adding lexical knowledge to the rule system, even when this knowledge is derived from a relatively small training set, significantly improves performance, and also that more training data leads to greater improvements.

Model	Training	Precision	Recall	F-measure
Surface String Measure				
Baseline	-	0.56	1.00	0.72
all-arg	10K	0.64	0.92	0.76
animacy	10K	0.62	0.93	0.75
all-arg	**FullWiki**	**0.68**	**0.95**	**0.81**
animacy	FullWiki	0.70	0.92	0.79
Full NL pipeline				
Baseline	-	0.65	0.94	0.77
all-arg	10K	0.75	0.88	0.81
animacy	10K	0.73	0.90	0.80
all-arg	**FullWiki**	**0.78**	**0.90**	**0.84**
animacy	FullWiki	0.81	0.88	0.84

Table 5: Performance on deverbal nouns with one possessive argument

5.1.2 Error Analysis and Discussion

We looked at the errors produced by the best-performing model, *all-arg* trained on the FullWiki set. There were 49 false negatives (i.e. cases where the human judge decided that the underlying relationship between the deverbal and its argument is ‘subject’, but our system produced a different relation or no relation at all), covering 39 unique deverbal nouns. Of these, 20 deverbal nouns were predicted by the model to prefer objects (e.g., *Hopkins’ accusation*), and 19 did not get assigned either subject or object due to other errors (including a few mislabeled evaluation sentences).

Some of the false negatives involved deverbal nouns that refer to reciprocal predicates such as *his marriage*, or causative ones such as *Berlin’s unity*, which could map to subject or objects. Our current system does not allow us to express such ambiguity, but it is a possible future improvement.

Looking at the false negatives produced by the *all-arg* model, 3 deverbal nouns received more accurate predictions with the *animacy* model (e.g., *his sight*; *Peter Kay’s statement*). Intuitively, the *animacy* model should in general make more informed decisions about the argument mappings because it takes properties of individual arguments into account. However, as we have seen, it does not in fact outperform the model that treats all arguments the same way.

We believe this is due to the fact that the animacy model was trained on less data than the *all-arg* model, because we only considered animate pronouns and common nouns when generating argument-mapping predictions. Excluding all named entities and non-animate pronouns most likely had an effect on the number of deverbals for which the model was able to make accurate predictions. In the next iteration, we would like to use all available arguments, relying on the named entity type and information available in WordNet for common nouns to distinguish between animate and inanimate arguments.

The *all-arg* model evaluation resulted in 131 false positives (cases where the model predicted the relation to be ‘subject’, but the human judge thought it was something else). Of these, 105 were marked by the human judge as having objects, 8 as having a verbal relation other than subject or object, 9 as having nominal modifiers, 9 has having no deverbal.

Altogether, false positives covered 85 unique verbs. Of these, 48 had been explicitly predicted by our model to prefer subjects, and the rest had no explicit prediction, thus defaulting to having a subject. 3 of these deverbals would have been correctly identified as having objects by the *animacy* model (e.g., *his composition*; *her representation*).

Although it is hard to predict the outcome of a statistical model, we feel that more reliable information about the animacy of arguments at training time would improve the performance of the *animacy* model, potentially making it better than the *all-arg* model.

5.2 ‘Of’ Arguments of Deverbal Nouns

The evaluation procedure for ‘of’ arguments was the same as for possessive arguments, except that the default argument mapping was ‘object’, and the evaluated decision was whether a particular role was object or non-object. Ignoring sentence with erroneous parses, we had 677 evaluation examples.

5.2.1 Results

Results for all models are summarized in Table 6. All models outperformed the baseline on all training sets and on both the surface or full-pipeline measures.

As with possessive arguments, the *all-arg* and *animacy* models performed about the same, with both the FullWiki and 10K training sets.

The 10K-trained *animacy* model did not do as poorly as might have been expected given its low prediction rate for deverbals with animate arguments in our evaluation set. The better-than-expected performance may be explained by low incidence of animate arguments in this set.

Model	Training	Precision	Recall	F-measure
Surface String Measure				
Baseline	-	0.60	1.00	0.75
all-arg	10K	0.68	0.97	0.80
animacy	10K	0.66	0.94	0.78
all-arg	**FullWiki**	**0.71**	**0.97**	**0.82**
animacy	FullWiki	0.70	0.91	0.79
Full NL pipeline				
Baseline	-	0.61	0.89	0.73
all-arg	10K	0.71	0.86	0.78
animacy	10K	0.70	0.85	0.77
all-arg	**FullWiki**	**0.78**	**0.87**	**0.82**
animacy	FullWiki	0.80	0.85	0.82

Table 6: Performance on deverbal nouns with one ‘of’ argument

5.2.2 Error Analysis and Discussion

We looked at the errors produced by the best-performing model, *all-arg* trained on the FullWiki set. There were 53 false negatives (cases where the human judged marked the relation as 'object' but the system marked it as something else), covering 42 unique deverbal nouns. Of these 7 were (incorrectly) predicted by the model to prefer subjects (e.g., *operation of a railway engine*), and the rest were misidentified due to other errors.

There were 101 false positives (cases where the system marked the role as object, but the human judge disagreed). Of these, the human judged marked 54 as subject, 21 as other verbal role, 13 as nominal modifier, and 13 as non-deverbal.

Of the 72 unique deverbals in the false-positive set, our model incorrectly predicted that 38 should prefer objects (such as *Adoration of the Magi*; *under the direction of Bishop Smith*)). For 30 deverbals, the model made no prediction, and the default mapping to object turned out to be incorrect. It is unclear to what extent better information about animacy would have helped.

6 Related Work

One of the earliest computational attempts to derive argument structures for deverbal nouns is (Hull and Gomez, 1996), with hand-crafted mapping rules for a small set of individual nouns, exemplifying a highly precise but not easily scalable method.

In recent years, NomBank (Meyers et al., 2004a) has provided a set of about 200,000 manually annotated instances of nominalizations with arguments, giving rise to supervised machine-learned approaches such as (Pradhan et al., 2004) and (Liu and Ng, 2007), which perform fairly well in the overall task of classifying deverbal arguments. However, no evaluation results are provided for specific, problematic classes of nominal arguments such as possessives; it is likely that the amount of annotations in NomBank is insufficient to reliably map such cases onto verbal arguments.

(Padó et al., 2008) describe an unsupervised approach that, like ours, uses verbal argument patterns to deduce deverbal patterns, though the resulting labels are semantic roles used in SLR tasks (cf. (Gildea and Jurafsky, 2000)) rather than syntactic roles. A combination of our much larger training set and the sophisticated probabilistic methods used by Padó et al. would most likely improve performance for both syntactic and semantic roles labelling tasks.

7 Conclusions and Future Work

We have demonstrated that large amounts of lexical data derived from an unsupervised parsed corpus improve role assignment for deverbal nouns. The improvements are significant even with a relatively small training set, relying on parses that have not been hand-corrected, using a very simple prediction model. Larger amounts of extracted data improve performance even more.

There is clearly still headroom for improvement in this method. In a pilot study, we used argument preferences for individual deverbal-argument pairs, falling back to deverbal-only generalizations when more specific patterns were not available. This model had slightly higher precision and slightly lower recall than the deverbal-only model, suggesting that a more sophisticcated probabilistic prediction model may be needed.

In addition, performance should improve if we allow non-binary decisions: in addition to mapping deverbal arguments to subject or object of the underlying verb, we could allow mappings such as "unknown" or "ambiguous". The same training sets can be used to produce a model that makes a 3- or 4-way split. In the possessive and 'of' sets, the "unknown / ambiguous" class would cover between 15% and 20% of all the data. This third possibility becomes even more important for other deverbal arguments. For example, if the deverbal noun has a prenominal modifier (as in *city destruction*), in a third of the cases the underlying relation is neither the subject nor the object (Lapata, 2002).

And, of course, the methodology of extracting lexical preferences based on large parsed corpora can be applied to many other NL tasks not related to deverbal nouns.

Acknowledgments

We gratefully acknowledge the helpful advice and comments of our colleagues Tracy Holloway King and Dick Crouch, as well as the three anonymous reviewers.

References

Michele Banko and Eric Brill. 2001. Scaling to very very large corpora for natural language disambiguation. In *ACL*, pages 26–33.

Richard S. Crouch and Tracy Holloway King. 2006. Semantics via f-structure rewriting. In *Proceedings of the Lexical Functional Grammar Conference 2006*.

Dick Crouch, Mary Dalrymple, Ron Kaplan, Tracy Holloway King, John Maxwell, and Paula Newman. 2009. XLE Documentation. Available On-line.

D. Cutting et al. Apache Hadoop Project. http://hadoop.apache.org/.

Mary Dalrymple. 2001. *Lexical Functional Grammar*. Academic Press. Syntax and Semantics, volume 34.

Christiane Fellbaum, editor. 1998. *WordNet: An Electronic Lexical Database*. The MIT Press.

Daniel Gildea and Daniel Jurafsky. 2000. Automatic labeling of semantic roles. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics*, pages 512–520, Hong Kong, October. Association for Computational Linguistics.

Olga Gurevich, Richard Crouch, Tracy Holloway King, and Valeria de Paiva. 2008. Deverbal nouns in knowledge representation. *Journal of Logic and Computation*, 18:385–404.

Richard D. Hull and Fernando Gomez. 1996. Semantic interpretation of nominalizations. In *AAAI/IAAI, Vol. 2*, pages 1062–1068.

Ronald Kaplan and John T. Maxwell. 1995. A method for disjunctive constraint satisfaction. In *Formal Issues in Lexical-Functional Grammar*. CSLI Press.

Judith Klavans and Philip Resnik, editors. 1996. *The Balancing Act. Combining Symbolic and Statistical Approaches to Language*. The MIT Press.

Maria Lapata. 2002. The disambiguation of nominalizations. *Computational Linguistics*, 28(3):357–388.

Chang Liu and Hwee Tou Ng. 2007. Learning predictive structures for semantic role labeling of nombank. In *ACL*.

Catherine Macleod, Ralph Grishman, Adam Meyers, Leslie Barrett, and Ruth Reeves. 1998. NOMLEX: A lexicon of nominalizations. In *Proceedings of EURALEX'98*.

Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of english: The penn treebank. *Computational Linguistics*, 19(2):313–330.

A. Meyers, R. Reeves, C. Macleod, R. Szekely, V. Zielinska, B. Young, and R. Grishman. 2004a. The nombank project: An interim report. In A. Meyers, editor, *HLT-NAACL 2004 Workshop: Frontiers in Corpus Annotation*, pages 24–31, Boston, Massachusetts, USA, May 2 - May 7. Association for Computational Linguistics.

Adam Meyers, Ruth Reeves, Catherine Macleod, Rachel Szekely, Veronika Zielinska, Brian Young, and Ralph Grishman. 2004b. The cross-breeding of dictionaries. In *Proceedings of LREC-2004*.

Mary Nunes. 1993. Argument linking in english derived nominals. In Robert Van Valin, editor, *Advances in Role and Reference Grammar*, pages 375–432. John Benjamins.

Sebastian Padó, Marco Pennacchiotti, and Caroline Sporleder. 2008. Semantic role assignment for event nominalisations by leveraging verbal data. In *Proceedings of CoLing08*.

Sameer Pradhan, Honglin Sun, Wayne Ward, James H. Martin, and Daniel Jurafsky. 2004. Parsing arguments of nominalizations in english and chinese. In Daniel Marcu Susan Dumais and Salim Roukos, editors, *HLT-NAACL 2004: Short Papers*, pages 141–144, Boston, Massachusetts, USA, May 2 - May 7. Association for Computational Linguistics.

Randolph Quirk, Sidney Greenbaum, Geoffrey Leech, and Jan Svartvik. 1985. *A Comprehensive Grammar of the English Language*. Longman.

Stefan Riezler, Tracy Holloway King, Ronald Kaplan, John T. Maxwell II, Richard Crouch, and Mark Johnson. 2002. Parsing the Wall Street Journal using a Lexical-Functional Grammar and discriminative estimation techniques. In *Proceedings of the ACL'02*.

Scott A. Waterman. 2009. Distributed parse mining. In *Software engineering, testing, and quality assurance for natural language processing (SETQA-NLP 2009)*.