INTERNATIONAL COMPUTER SCIENCE INSTITUTE

1947 Center St. • Suite 600 • Berkeley, California 94704-1198 • (510) 666-2900 • FAX (510) 666-2956

Automatic Labeling of Semantic Roles

Daniel Gildea* **and Daniel Jurafsky†**

TR-01-005

May 2001

Abstract

We present a system for identifying the semantic relationships, or *semantic roles*, filled by constituents of a sentence within a semantic frame. Given an input sentence, the system labels constituents with either abstract semantic roles such as AGENT or PATIENT, or more domain-specific semantic roles such as SPEAKER, MESSAGE, and TOPIC.

The system is based on statistical classifiers which were trained on 653 semantic role types from roughly 50,000 sentences. Each sentence had been hand-labeled with semantic roles in the FrameNet semantic labeling project. We then parsed each training sentence and extracted various lexical and syntactic features, including the syntactic category of the constituent, its grammatical function, and position in the sentence. These features were combined with knowledge of the target verb, noun, or adjective, as well as information such as the prior probabilities of various combinations of semantic roles. We also used various lexical clustering algorithms to generalize across possible fillers of roles. Test sentences were parsed, were annotated with these features, and were then passed through the classifiers.

Our system achieves 82% accuracy in identifying the semantic role of pre-segmented constituents. At the harder task of simultaneously segmenting constituents and identifying their semantic role, the system achieved 65% precision and 61% recall.

Our study also allowed us to compare the usefulness of different features and feature-combination methods in the semantic role labeling task.

*University of California, Berkeley, and International Computer Science Institute

†University of Colorado, Boulder

1 Introduction

Recent years have been exhilarating ones for natural language understanding. The excitement and rapid advances that had characterized other language processing tasks like speech recognition, part of speech tagging, and parsing, have finally begun to appear in tasks in which understanding and semantics play a greater role. For example, there has been widespread commercial deployment of simple speech-based natural language understanding systems which answer questions about flight arrival times, give directions, report on bank balances or perform simple financial transactions. More sophisticated research systems can generate concise summaries of news articles, answer fact-based questions, and recognize complex semantic and dialog structure.

But the challenges that lie ahead are still similar to the challenge that the field has faced since Winograd (1972): moving away from carefully hand-crafted, domain-dependent systems toward robustness and domain-independence. This goal is not as far away as it once was, thanks to the development of large semantic databases like WordNet (Fellbaum, 1998), and of general-purpose domain-independent algorithms like named-entity recognition.

Current information extraction and dialogue understanding systems, however, are still based on domain-specific frame-and-slot templates. Systems for booking airplane information are based on domain-specific frames with slots like FROM_AIRPORT, TO_AIRPORT, or DEPART_TIME. Systems for studying mergers and acquisitions are based on slots like JOINT_VENTURE_COMPANY, PRODUCTS, RELATIONSHIP, and AMOUNT. In order for natural language understanding tasks to proceed beyond these specific domains, we need semantic frames and a semantic understanding system which don't require a new set of slots for each new application domain.

In this paper we describe a shallow semantic interpreter based on semantic roles that are less domain-specific than TO_AIRPORT or JOINT_VENTURE_COMPANY. These roles are defined at the level semantic frames (see (Fillmore, 1976) for a description of frame-based semantics), which describe abstract actions or relationships along their participants.

For example, the JUDGEMENT frame contains roles like JUDGE, EVALUEE, and REASON, while the STATEMENT frame contains roles like SPEAKER, ADDRESSEE, and MESSAGE, as the following examples show:

(1) [$_{Judge}$ She] **blames** [$_{Evaluee}$ the Government] [$_{Reason}$ for failing to do enough to help] .

(2) [$_{Message}$ "I'll knock on your door at quarter to six"] [$_{Speaker}$ Susan] **said**.

These shallow semantic roles could play an important role in information extraction, for example allowing a system to determine that in the sentence "The first one crashed" the syntactic subject is the VEHICLE, but in the sentence "The first one crashed it" the syntactic subject is the AGENT. But this shallow semantic level of interpretation can be used for many purposes besides generalizing information extraction and semantic dialogue systems. One such application is in word-sense disambiguation, where the roles associated with a word can be cues to its sense. For example, Lapata and Brew (1999) and others have shown that the different syntactic subcategorization frames of a verb like "serve" can be used to help disambiguate a particular instance of the word "serve". Adding semantic role subcategorization information to this syntactic information could extend this idea to use richer semantic knowledge. Semantic roles could also act as an important intermediate representation in statistical machine translation or automatic text summarization and in the emerging field of Text Data Mining (TDM) (Hearst, 1999). Finally, incorporating semantic roles into probabilistic models of language may yield more accurate parsers and better language models for speech recognition.

This paper describes an algorithm for identifying the semantic roles filled by constituents in a sentence. We apply statistical techniques that have been successful for the related problems of syntactic parsing, part of speech tagging, and word sense disambiguation, including probabilistic parsing and statistical classification. Our statistical algorithms are trained on a hand-labeled dataset:

the FrameNet database (Baker, Fillmore, and Lowe, 1998). The FrameNet database defines a tagset of semantic roles called **frame elements**, and includes roughly 50,000 sentences from the British National Corpus which have been hand-labeled with these frame elements.

We present our system in stages, beginning in the Section 2 with a description of the task and the set of frame elements/semantic roles used, and continuing in Section 3 by relating the system to previous research. We divide the problem of labeling roles into two parts: finding the relevant sentence constituents, and giving them the correct labels. In Section 4, we explore how to choose the labels when the boundaries are known, and in Section 5 we return to the problem of identifying the sentence parts to be labeled. Section 6 examines how the choice of the set of semantic roles affects results. Section 7 compares various strategies for improving performance by generalizing across lexical statistics for role fillers, and Section 8 examines representations of sentence-level argument structure. Finally, we draw conclusions and discuss future directions.

2 Semantic Roles

Semantic roles are probably one of the oldest classes of constructs in linguistic theory, dating back thousands of years to Panini's *kāraka* theory. Longevity, in this case, begets variety, and the literature records scores of proposals for sets of semantic roles. These sets of roles range from the very specific to the very general, and many have been used in computational implementations of one type or another.

At the specific end of the spectrum are domain-specific roles such as the FROM_AIRPORT, TO_AIRPORT, or DEP_TIME discussed above, or verb-specific roles like EATER and EATEN for the verb *eat*. The opposite end of the spectrum consists of theories with only two 'proto-roles' or 'macroroles': PROTO-AGENT and PROTO-PATIENT (Van Valin, 1993; Dowty, 1991). In between lie many theories with around ten roles or so, such as Fillmore (1971)'s list of nine: AGENT, EXPERIENCER, INSTRUMENT, OBJECT, SOURCE, GOAL, LOCATION, TIME, and PATH. [1]

Many of these sets of roles have been proposed either by linguists as part of theories of *linking*, the part of grammatical theory which describes the relationship between semantic roles and their syntactic realization, or by computer scientists as part of implemented natural language understanding systems. As a rule, the more abstract roles have been proposed by linguists, who are more concerned with explaining generalizations across verbs in the syntactic realization of their arguments, while the more specific roles are more often proposed by computer scientists, who are more concerned with the details of the realization of the arguments of single verbs.

The FrameNet project proposes roles which are neither as general as the ten abstract thematic roles, nor as specific as the thousands of potential verb-specific role. FrameNet roles are defined for each semantic frame. A frame is a schematic representations of situations involving various participants, props, and other conceptual roles (Fillmore, 1976). For example, the frame CONVERSATION, shown in Figure 1, is invoked by the semantically related verbs "argue", "banter", "debate", "converse", and "gossip" as well as the nouns "argument", "dispute", "discussion" and "tiff", and is defined as follows:

(3) Two (or more) people talk to one another. No person is construed as only a speaker or only an addressee. Rather, it is understood that both (or all) participants do some speaking and some listening—the process is understood to be symmetrical or reciprocal.

The roles defined for this frame, and shared by all its lexical entries, include PROTAGONIST1 and PROTAGONIST2 or simply PROTAGONISTS for the participants in the conversation, as well as MEDIUM, and TOPIC. Similarly, the JUDGMENT frame mentioned above has the roles JUDGE, EVALUEE, and REASON, and is invoked by verbs like "blame", "admire", and "praise", and nouns

[1] There are scores of other theories with slightly different sets of roles, including, among many others, (Fillmore, 1968), (Jackendoff, 1972), (Schank, 1972); see (Somers, 1987) for an excellent summary.

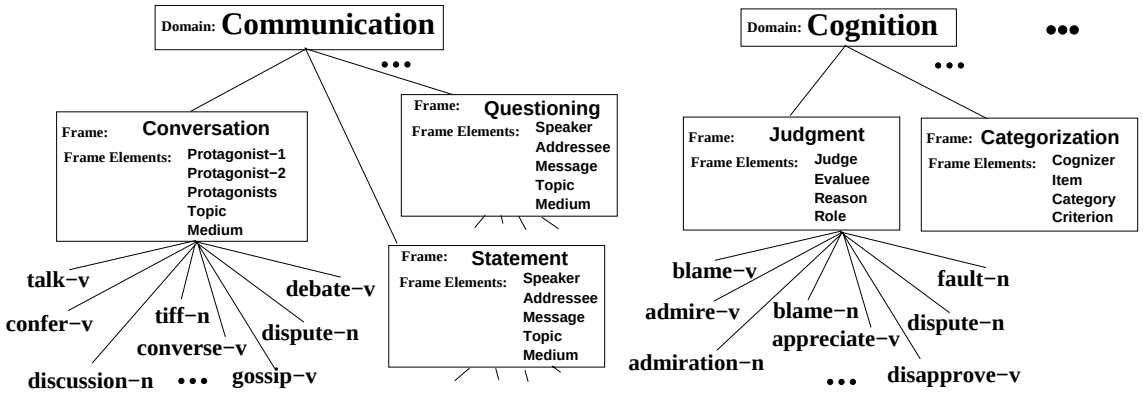

Figure 1: Sample domains and frames from the FrameNet lexicon.

like "fault" and "admiration". A number of annotated examples from the JUDGMENT frame are included below to give a flavor of the FrameNet database:

(4) [$_{Judge}$ She] **blames** [$_{Evaluee}$ the Government] [$_{Reason}$ for failing to do enough to help] .

(5) Holman would characterise this as **blaming** [$_{Evaluee}$ the poor] .

(6) The letter quotes Black as saying that [$_{Judge}$ white and Navajo ranchers] misrepresent their livestock losses and **blame** [$_{Reason}$ everything] [$_{Evaluee}$ on coyotes] .

(7) The only dish she made that we could tolerate was [$_{Evaluee}$ syrup tart which] [$_{Judge}$ we] **praised** extravagantly with the result that it became our unhealthy staple diet.

(8) I 'm bound to say that I meet a lot of [$_{Judge}$ people who] **praise** [$_{Evaluee}$ me] [$_{Reason}$ for speaking up] but don't speak up themselves.

(9) Specimens of her verse translations of Tasso Jerusalem Delivered and Verri Roman Nights circulated to [$_{Manner}$ warm] [$_{Judge}$ critical] **praise** but unforeseen circumstance prevented their publication.

(10) And if Sam Snort hails Doyler as monumental is he perhaps erring on the side of being excessive in [$_{Judge}$ his] **praise**.

Defining semantic roles at this intermediate frame level may avoid some of the well-known difficulties of defining a unique small set of universal, abstract thematic roles, while also allowing some generalization across the roles of different verbs, nouns, and adjectives, each of which adds additional semantics to the general frame, or highlights a particular aspect of the frame. One way of thinking about very abstract thematic roles in a FrameNet systems is as frame elements which are defined in very abstract frames such as "action" and "motion", at the top of an inheritance hierarchy of semantic frames (Fillmore and Baker, 2000).

The examples above illustrate another difference between frame elements and thematic roles, at least as commonly implemented. Where thematic roles tend to be arguments mainly of verbs, frame elements can be arguments of any predicate, and the FrameNet database thus includes nouns and adjectives as well as verbs.

The examples above also illustrate a few of the phenomena that make it hard to automatically identify frame elements. Many of these are caused by the fact that there is not always a direct correspondence between syntax and semantics. While the subject of **blame** is often the JUDGE, the direct object of **blame** can be an EVALUEE (e.g., 'the poor' in "blaming the poor") or a REASON

(e.g., 'everything' in "blame everything on coyotes"). The JUDGE can also be realized as a genitive pronoun, (e.g. 'his' in "his praise") or even an adjective (e.g. 'critical' in "critical praise").

The preliminary version of the FrameNet corpus used for our experiments contained 67 frame types from 12 general semantic domains chosen for annotation. A complete list of the domains is shown in Table 1, along with representative frames and predicates. Within these frames, examples of a total of 1462 distinct lexical predicates, or **target words**, were annotated: 927 verbs, 339 nouns, and 175 adjectives. There are a total of 49,013 annotated sentences, and 99,232 annotated frame elements (which do not include the target words themselves).

Domain	*Sample Frames*	*Sample Predicates*
Body	Action	flutter, wink
Cognition	Awareness	attention, obvious
	Judgment	blame, judge
	Invention	coin, contrive
Communication	Conversation	bicker, confer
	Manner	lisp, rant
Emotion	Directed	angry, pleased
	Experiencer	bewitch, rile
General	Imitation	bogus, forge
Health	Response	allergic, susceptible
Motion	Arriving	enter, visit
	Filling	annoint, pack
Perception	Active	glance, savour
	Noise	snort, whine
Society	Leadership	emperor, sultan
Space	Adornment	cloak, line
Time	Duration	chronic, short
	Iteration	daily, sporadic
Transaction	Basic	buy, spend
	Wealthiness	broke, well-off

Table 1: Semantic domains with sample frames and predicates from the FrameNet lexicon

3 Related Work

Assignment of semantic roles is an important part of language understanding, and has been attacked by many computational systems. Traditional parsing and understanding systems, including implementations of unification-based grammars such as HPSG (Pollard and Sag, 1994), rely on hand-developed grammars which must anticipate each way in which semantic roles may be realized syntactically. Writing such grammars is time-consuming, and typically such systems have limited coverage.

Data-driven techniques have recently been applied to template-based semantic interpretation in limited domains by "shallow" systems that avoid complex feature structures, and often perform only shallow syntactic analysis. For example, in the context of the Air Traveler Information System (ATIS) for spoken dialogue, Miller et al. (1996) computed the probability that a constituent such as "Atlanta" filled a semantic slot such as DESTINATION in a semantic frame for air travel. In a data-driven approach to information extraction, Riloff (1993) builds a dictionary of patterns for filling slots in a specific domain such as terrorist attacks, and Riloff and Schmelzenbach (1998) extend this technique to automatically derive entire case frames for words in the domain. These last systems make use of a limited amount of hand labor to accept or reject automatically generated hypotheses.

They show promise for a more sophisticated approach to generalize beyond the relatively small number of frames considered in the tasks. More recently, a domain independent system has been trained by Blaheta and Charniak (2000) on the function tags such as MANNER and TEMPORAL included in the Penn Treebank corpus. Some of these tags correspond to FrameNet semantic roles, but the Treebank tags do not include all the arguments of most predicates. In this work, we aim to develop a statistical system to automatically learn to identify all the semantic roles for a wide variety of predicates in unrestricted text.

4 Probability Estimation for Roles

We divide the task of labeling frame elements into two subtasks: that of identifying the boundaries of the frame elements in the sentences, and that of labeling each frame element, given its boundaries, with the correct role. We first give results for a system which labels roles using human-annotated boundaries, returning to the question of automatically identifying the boundaries in Section 5.

4.1 Features Used in Assigning Semantic Roles

The system is a statistical one, based on training a classifier on a labeled training set, and testing on a held-out portion of the data. The system is trained by first using an automatic syntactic parser to analyze the 36,995 training sentences, matching annotated frame elements to parse constituents, and extracting various features from the string of words and the parse tree. During testing, the parser is run on the test sentences and the same features extracted. Probabilities for each possible semantic role r are then computed from the features. The probability computation will be described in the next section; here we discuss the features used.

The features used represent various aspect of the syntactic structure of the sentence as well as lexical information. The relationship between such surface manifestations and semantic roles is the subject of **linking theory** — see Levin and Hovav (1996) for a synthesis of work in this area. In general, linking theory argues that the syntactic realization of arguments of a predicate is predictable from semantics — exactly how this relationship works is the subject of much debate. Regardless of the underlying mechanisms used to generate syntax from semantics, the relationship between the two suggests that it may be possible to learn to recognize semantic relationships from syntactic cues, given examples with both types of information.

4.1.1 Phrase Type

Different roles tend to be realized by different syntactic categories. For example, in communication frames, the SPEAKER is likely to appear as a noun phrase, TOPIC as a prepositional phrase or noun phrase, and MEDIUM as a prepositional phrase, as in: "We talked about the proposal over the phone."

The phrase type feature we used indicates the syntactic category of the phrase expressing the semantic roles, using the set of syntactic categories of the Penn Treebank project, as described in Marcus, Santorini, and Marcinkiewicz (1993). In our data, frame elements are most commonly expressed as noun phrases (NP, 47% of frame elements in the training set), and prepositional phrases (PP, 22%). The next most common categories are adverbial phrases (ADVP, 4%), particles (e.g. "make something *up*" – PRT, 2%) and sentential clauses (SBAR, 2% and S 2%).

We used the parser of Collins (1997), a statistical parser trained on examples from the Penn Treebank, to generate parses of the same format for the sentences in our data. Phrase types were derived automatically from parse trees generated by the parser, as shown in Figure 2. Given the automatically generated parse tree, the constituent spanning each set of words annotated as a frame element was found, and the constituent's nonterminal label was taken as the phrase type.

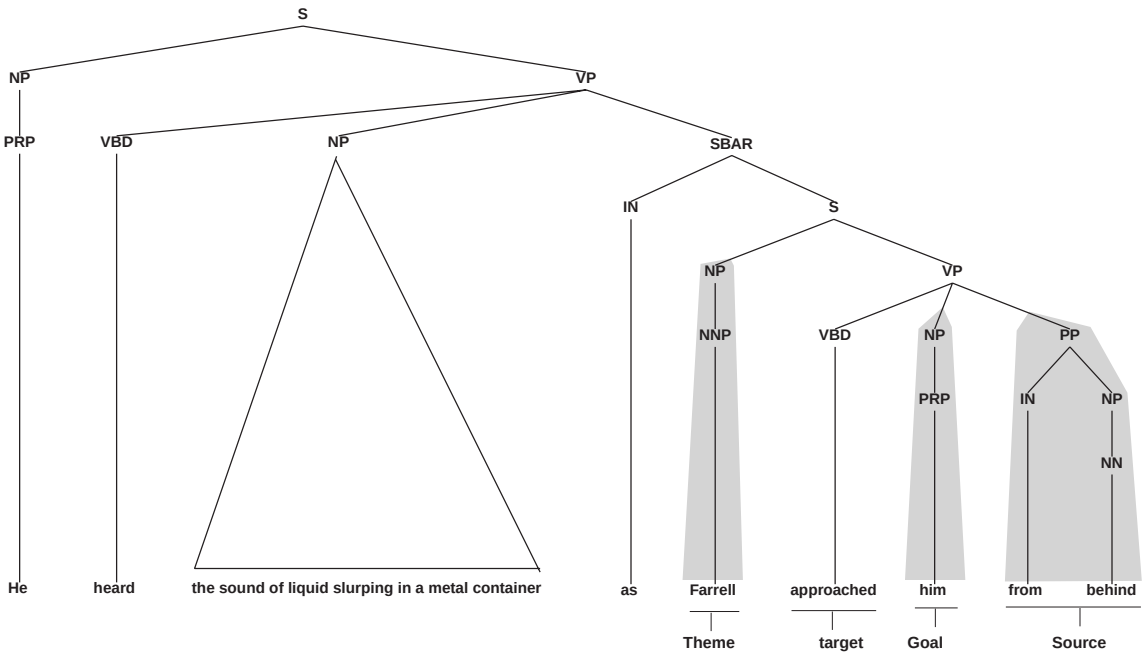

Figure 2: A sample sentence with parser output (above) and FrameNet annotation (below). Parse constituents corresponding to frame elements are highlighted.

The matching was performed by calculating the starting and ending word positions for each constituent in the parse tree, as well as for each annotated frame element, and matching each frame element with the parse constituent with the same beginning and ending points. Punctuation was ignored in this computation. Due to parsing errors, or, less frequently, mismatches between the parse tree formalism and the FrameNet annotation standards, there was sometimes no parse constituent matching an annotated frame element. 13% of the frame elements in the training set had no matching parse constituent. These cases were discarded during training; during testing, the largest constituent beginning at the frame element's left boundary and lying entirely within the element was used to calculate the features. This handles common parse errors such as a prepositional phrase being incorrectly attached to a noun phrase at the right hand edge, and it guarantees that some syntactic category will be returned: the part of speech tag of the frame element's first word in the limiting case.

4.1.2 Grammatical Function

The correlation between semantic roles and syntactic realization as subject or direct object is one of the primary facts that linking theory attempts to explain. It was a motivation for the case hierarchy of Fillmore (1968), which allowed such rules as "if there is an underlying AGENT, it becomes the syntactic subject". Similarly, in his theory of macroroles, Van Valin (1993) describes the ACTOR as being preferred in English for the subject. Functional grammarians consider syntactic subjects to have been historically grammaticalized agent markers. As an example of how this feature is useful, in the sentence "He drove the car over the cliff", the subject NP is more likely to fill the AGENT role than the other two NPs.

The grammatical function feature we used attempts to indicate a constituent's syntactic relation to the rest of the sentence, for example as a subject or object of a verb. As with phrase type, this feature was read from parse trees returned by the parser. After experimentation with various

versions of this feature, we restricted it to apply only to NPs, as it was found to have little effect on other phrase types. Only two values for this feature were used: *subject* and *object*. An NP node whose parent is an S node was assigned the function *subject*, and an NP whose parent is a VP was assigned the function *object*. In cases where the NP's immediate parent was neither an S or VP, the nearest S or VP ancestor was found, and the value of the feature assigned accordingly.

4.1.3 Position

In order to overcome errors due to incorrect parses, as well as to see how much can be done without parse trees, we introduced position as a feature. This feature simply indicates whether the constituent to be labeled occurs before or after the predicate defining the semantic frame. We expected this feature to be highly correlated with grammatical function, since subjects will generally appear before a verb, and objects after.

Although we do not have hand-checked parses against which to measure the performance of the automatic parser on our corpus, the result that 13% of frame elements have no matching parse constituent gives a rough idea of the parser's accuracy. Almost all of these cases are due to parser error. Other parser errors include cases where a constituent is found, but with the incorrect label or internal structure. This measure also considers only the individual constituent representing the frame element — the parse for the rest of the sentence may be incorrect, resulting in an incorrect value for the grammatical relation feature. Collins (1997) reports 88% labeled precision and recall on individual parse constituents on data from the Penn Treebank, roughly consistent with our finding of at least 13% error.

4.1.4 Voice

The distinction between active and passive verbs plays an important role in the connection between semantic role and grammatical function, since direct objects of active verbs correspond to subjects of passive verbs. From the parser output, verbs were classified as active or passive by building a set of 10 passive-identifying patterns. Each of the patterns requires both a passive auxiliary (some form of "to be" or "to get") and a past participle.

4.1.5 Head Word

As previously noted, we expected lexical dependencies to be extremely important in labeling semantic roles, as indicated by their importance in related tasks such as parsing. For example, in a communication frame, noun phrases headed by "Bill", "brother", or "he" are more likely to be the SPEAKER, while those headed by "proposal", "story", or "question" are more likely to be the TOPIC. (We did not attempt to resolve pronoun references.) Since the parser we used assigns each constituent a head word as an integral part of the parsing model, we were able to read the head words of the constituents from the parser output, using the same set of rules for identifying the head child of each constituent in the parse tree.

4.2 Probability Estimation

For our experiments, we divided the FrameNet corpus as follows: one-tenth of the annotated sentences for each target word were reserved as a test set, and another one-tenth were set aside as a tuning set for developing our system. A few target words with fewer than ten examples were removed from the corpus. In our corpus, the average number of sentences per target word is only 34, and the number of sentences per frame is 732 — both relatively small amounts of data on which to train frame element classifiers.

In order to automatically label the semantic role of a constituent, we wish to estimate a probability distribution telling us how likely the constituent is to fill each possible role given the the features described above and the predicate, or target word, t:

$P(r|h, pt, gf, position, voice, t)$

It would be possible to calculate this distribution directly from the training data by counting the number of times each role is seen with a combination of features, and dividing by the total number of times the combination of features is seen:

$$P(r|h, pt, gf, position, voice, t) = \frac{\#(r, h, pt, gf, position, voice, t)}{\#(h, pt, gf, position, voice, t)}$$

However, in many cases, we will never have seen a particular combination of features in the training data, and in others we will have seen the combination only a small number of times, providing a poor estimate of the probability. The fact that there are only about 30 training sentences for each target word, and that the head word feature in particular can take on a large number of values (any word in the language), contribute to the sparsity of the data. Although we expect our features to interact in various ways, we cannot train directly on the full feature set. For this reason, we built our classifier by combining probabilities from distributions conditioned on a variety of combinations of features.

Distribution	*Coverage*	*Accuracy*	*Performance*
$P(r\|t)$	100%	40.9%	40.9%
$P(r\|pt, t)$	92.5	60.1	55.6
$P(r\|pt, gf, t)$	92.0	66.6	61.3
$P(r\|pt, position, voice)$	98.8	57.1	56.4
$P(r\|pt, position, voice, t)$	90.8	70.1	63.7
$P(r\|h)$	80.3	73.6	59.1
$P(r\|h, t)$	56.0	86.6	48.5
$P(r\|h, pt, t)$	50.1	87.4	43.8

Table 2: Distributions Calculated for Semantic Role Identification: r indicates semantic role, pt phrase type, gf grammatical function, h head word, and t target word, or predicate.

Table 2 shows the probability distributions used in the final version of the system. *Coverage* indicates the percentage of the test data for which the conditioning event had been seen in training data. *Accuracy* is the proportion of covered test data for which the correct role is predicted, and *Performance*, which is the product of coverage and accuracy, is the overall percentage of test data for which the correct role is predicted. Accuracy is somewhat similar to the familiar metric of *precision* in that it is calculated over cases for which a decision is made, and performance is similar to *recall* in that it is calculated over all true frame elements. However, unlike a traditional precision/recall trade-off, these results have no threshold to adjust, and the task is a multi-way classification rather than a binary decision. The distributions calculated were simply the empirical distributions from the training data. That is, occurrences of each role and each set of conditioning events were counted in a table, and probabilities calculated by dividing the counts for each role by the total number of observations for each conditioning event. For example, the distribution $P(r|pt, t)$ was calculated as follows:

$$P(r|pt, t) = \frac{\#(r, pt, t)}{\#(pt, t)}$$

Some sample probabilities calculated from the training are shown in Table 3.

As can be seen from Table 2, there is a trade-off between more specific distributions, which have high accuracy but low coverage, and less specific distributions, which have low accuracy but high

$P(r\|pt, gf, t)$	*Count in training data*
$P(r = \text{AGT}\|pt = \text{NP}, gf = \text{Subj}, t = \text{abduct}) = .46$	6
$P(r = \text{THM}\|pt = \text{NP}, gf = \text{Subj}, t = \text{abduct}) = .54$	7
$P(r = \text{THM}\|pt = \text{NP}, gf = \text{Obj}, t = \text{abduct}) = 1$	9
$P(r = \text{AGT}\|pt = \text{PP}, t = \text{abduct}) = .33$	1
$P(r = \text{THM}\|pt = \text{PP}, t = \text{abduct}) = .33$	1
$P(r = \text{COTHM}\|pt = \text{PP}, t = \text{abduct}) = .33$	1
$P(r = \text{MANR}\|pt = \text{ADVP}, t = \text{abduct}) = 1$	1

Table 3: Sample probabilities for $P(r|pt, gf, t)$ calculated from training data for the verb *abduct*. The variable gf is only defined for noun phrases. The roles defined for the *removing* frame in the *motion* domain are: AGENT, THEME, COTHEME ("... had been abducted *with him*") and MANNER.

coverage. The lexical head word statistics, in particular, are valuable when data are available, but are particularly sparse due to the large number of possible head words. In order to combine the strengths of the various distributions, we combined them in various ways to obtain an estimate of the full distribution $P(r|h, pt, gf, position, voice, t)$.

The first combination method is linear interpolation, which simply averages the probabilities given by each of the distributions:

$$\begin{aligned} P(r|constituent) \quad = \quad & \lambda_1 P(r|t) + \lambda_2 P(r|pt, t) + \\ & \lambda_3 P(r|pt, gf, t) + \lambda_4 P(r|pt, position, voice) + \\ & \lambda_5 P(r|pt, position, voice, t) + \lambda_6 P(r|h) + \\ & \lambda_7 P(r|h, t) + \lambda_8 P(r|h, pt, t) \end{aligned}$$

where $\sum_i \lambda_i = 1$. The geometric mean, when expressed in the log domain, is similar:

$$\begin{aligned} P(r|constituent) \quad = \frac{1}{Z} exp\{ \quad & \lambda_1 logP(r|t) + \lambda_2 logP(r|pt, t) + \\ & \lambda_3 logP(r|pt, gf, t) + \lambda_4 logP(r|pt, position, voice) + \\ & \lambda_5 logP(r|pt, position, voice, t) + \lambda_6 logP(r|h) + \\ & \lambda_7 logP(r|h, t) + \lambda_8 logP(r|h, pt, t) \quad \} \end{aligned}$$

where Z is a normalizing constant ensuring that $\sum_r P(r|constituent) = 1$.

The results shown in Table 4 reflect equal values of λ for each distribution defined for the relevant conditioning event (but excluding distributions for which the conditioning event was not seen in the training data). A few other schemes for choosing the interpolation weights were tried, for example giving more weight to distributions for which more training data was available, as they might be expected to be more accurately estimated. However, this was found to have relatively little effect. We attribute this to the fact that the evaluation depends only on the ranking of the probabilities rather than their exact values.

In the "backoff" combination method, a lattice was constructed over the distributions in Table 2 from more specific conditioning events to less specific, as shown in Figure 3. The lattice is used to select a subset of the available distributions to combine. The less specific distributions were used only when no data was present for any more specific distribution. Thus, the distributions selected are arranged in a cut across the lattice representing the most specific distributions for which data is available. The selected probabilities were combined with both linear interpolation and a geometric mean.

Although this lattice is reminiscent of techniques of backing off to less specific distributions commonly used in n-gram language modeling, it differs in that we only use the lattice to select

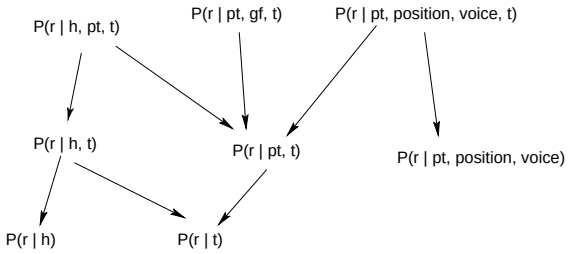

Figure 3: Lattice organization of the distributions from Table 2, with more specific distributions towards the top.

distributions for which the conditioning event has been seen in the training data. Discounting and deleted interpolation methods in language modeling typically are used to assign small, non-zero probability to a predicted variable unseen in the training data even when a specific conditioning even has been seen. In our case, we are perfectly willing to assign zero probability to a specific role (the predicted variable), because we are only interested in finding the role with the highest probability.

Combining Method	*Correct*
Linear Interpolation	79.5%
Geometric Mean	79.6
Backoff, linear interpolation	80.4
Backoff, geometric mean	79.6
Baseline: Most common role	40.9

Table 4: Results on Development Set, 8167 observations

	Linear Backoff	*Baseline*
Development Set	80.4%	40.9%
Test Set	76.9	40.6%

Table 5: Results on Test Set, using backoff linear interpolation system. The test set consists of 7900 observations.

The final system performed at 80.4% accuracy, which can be compared to the 40.9% achieved by always choosing the most probable role for each target word, essentially chance performance on this task. Results for this system on test data, held out during development of the system, are shown in Table 5.

4.3 Multiple Estimates of Grammatical Function

It is interesting to note that looking at a constituent's position relative to the target word along with active/passive information performed as well as reading grammatical function off the parse tree. A system using grammatical function, along with the head word, phrase type, and target word, but no passive information, scored 79.2%, compared with 80.4% for the full system. A similar system using position rather than grammatical function scored 78.8% — nearly identical performance. However, using head word, phrase type, and target word without either position or grammatical function yielded only 76.3%, indicating that while the two features accomplish a similar goal, it is important to include some measure of the constituent's syntactic relationship to the target word.

Our final system incorporated both features, giving a further, though not significant, improvement. As a guideline for interpreting these results, with 8167 observations, the threshold for statistical significance with $p < .05$ is a 1.0% absolute difference in performance.

Use of the active/passive feature made a further improvement: our system using position but no grammatical function or passive information scored 78.8%. Using position and passive information, but no grammatical function, brought performance to 80.5%. (We consider this identical to the 80.4% achieved with all features, shown in Tables 4 and 5, and prefer to leave features in the system in the case of equal performance.) Roughly 5% of the examples were identified as passive uses.

5 Identification of Frame Element Boundaries

The experiments described above have used human annotated frame element boundaries — here we address how well the frame elements can be found automatically. Experiments were conducted using features similar to those described above to identify constituents in a sentence's parse tree that were likely to be frame elements. However, the system is still given the human-annotated target word and the frame to which it belongs as inputs. We defer for now the task of identifying which frames come into play in a sentence, but envision that existing word sense disambiguation techniques could be applied to the task.

Our approach to finding frame elements is similar to the approach described in the previous section for labeling them: features are extracted from the sentence and its parse, and used to calculated probability tables, with the predicted variable, *fe*, being a binary indicator of whether a given constituent in the parse tree is or is not a frame element.

We introduce one new feature for this purpose: the **path** from the target word through the parse tree to the constituent in question, represented as a string of parse tree nonterminals linked by symbols indicating upward or downward movement through the tree, as shown in Figure 4. [2]

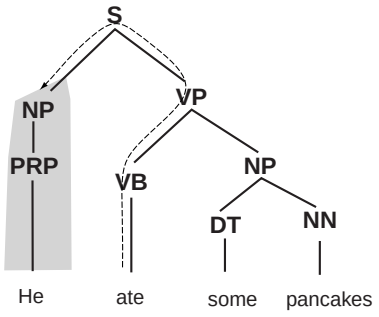

Figure 4: In this example, the **path** from the frame element "He" to the target word "ate" can be represented as VB ↑ VP ↑ S ↓ NP, with ↑ indicating upward movement in the parse tree and ↓ downward movement.

The other features used were the identity of the target word and the identity of the constituent's head word. The probability distributions calculated from the training data were $P(fe|path)$, $P(fe|path, t)$, and $P(fe|h, t)$, where *fe* indicates an event where the parse constituent in question is a frame element, *path* the path through the parse tree from the target word to the parse constituent, t the identity of the target word, and h the head word of the parse constituent. Some sample values from these distributions are shown in Table 6. For example, the path VB ↑ VP↓ NP, which corresponds

[2] This feature can be thought of as a variant of the grammatical function feature described in Section 4.1. Although experiments showed the distinction between the features to be of little importance for role labeling, the grounding of the path feature in the target word is crucial for frame element identification. The previous feature identifies all subjects, regardless of which verb's subject they are.

to the direct object of a verbal target word, had a high probability of being a frame element. The table also illustrates cases of sparse data for various feature combinations.

By varying the probability threshold at which a decision is made, one can plot a precision/recall curve as shown in Figure 5. $P(\text{fe}|path, t)$ performs relatively poorly due to fragmentation of the training data (recall only about 30 sentences are available for each target word). While the lexical statistic $P(fe|h, t)$ alone is not useful as a classifier, using it in linear interpolation with the path statistics improves results. The "interpolation" curve in Figure 5 reflects a linear interpolation of the form:

$$P(fe|p, h, t) = \lambda_1 P(fe|p) + \lambda_2 P(fe|p, t) + \lambda_3 P(fe|h, t) \tag{11}$$

Note that this method can only identify frame elements that have a corresponding constituent in the automatically generated parse tree. For this reason, it is interesting to calculate how many true frame elements overlap with the results of the system, relaxing the criterion that the boundaries must match exactly. Results for partial matching are shown in Table 7. Three types of overlap are possible: the identified constituent entirely within true frame element, the true frame element entirely within identified constituent, and neither sequence entirely within the other. An example of the first case is shown in Figure 6, where the true MESSAGE frame element is "Mandarin by a head", but due to an error in the parser output, no constituent exactly matches the frame elements boundaries. In this case, the system identifies two frame elements, indicated by shading, which together span the true frame element.

Distribution	Sample Prob.	*Count in training data*
$P(fe\|path)$	$P(fe\|path =$VBD↑VP↓ADJP↓ADVP$) = 1$	1
	$P(fe\|path =$VBD ↑ VP↓ NP$) = .73$	3963
	$P(fe\|path =$VBN ↑ VP↓ NP↓ PP↓ S$) = 0$	22
$P(fe\|path, t)$	$P(fe\|path =$JJ↑ADJP↓PP$, t =$apparent$) = 1$	10
	$P(fe\|path =$NN↑NP↑PP↑VP↓PP$, t =$departure$) = .4$	5
$P(fe\|h, t)$	$P(fe\|h =$sudden$, t =$apparent$) = 0$	2
	$P(fe\|h =$to$, t =$apparent$) = .11$	93
	$P(fe\|h =$that$, t =$apparent$) = .21$	81

Table 6: Sample probabilities for a constituent being a frame element.

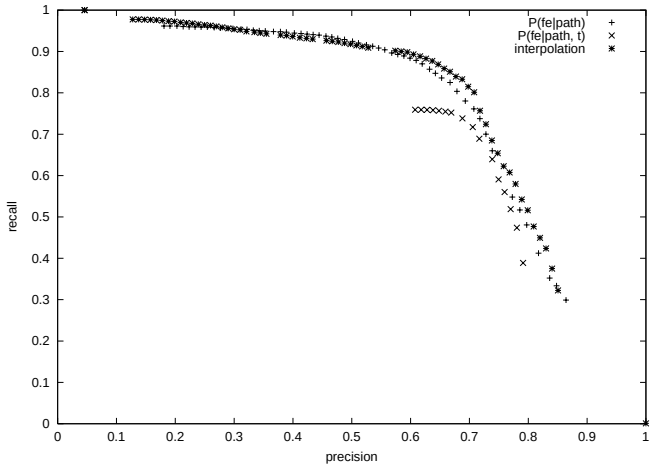

Figure 5: Precision/Recall plot for various methods of identifying frame elements. Recall is calculated over only frame elements with matching parse constituents.

Type of Overlap	*Identified Constituents*	*Number*
Exactly matching boundaries	66%	5421
Identified constituent entirely within true frame element	8	663
True frame element entirely within identified constituent	7	599
Neither entirely within the other	0	26
No overlap with any true frame element	13	972

Table 7: Results on Identifying Frame Elements (FEs), including partial matches. Results obtained using $P(fe|path)$ with threshold at .5. A total of 7681 constituents were identified as FEs, 8167 FEs were present in hand annotations, of which matching parse constituents were present for 7053 (86%).

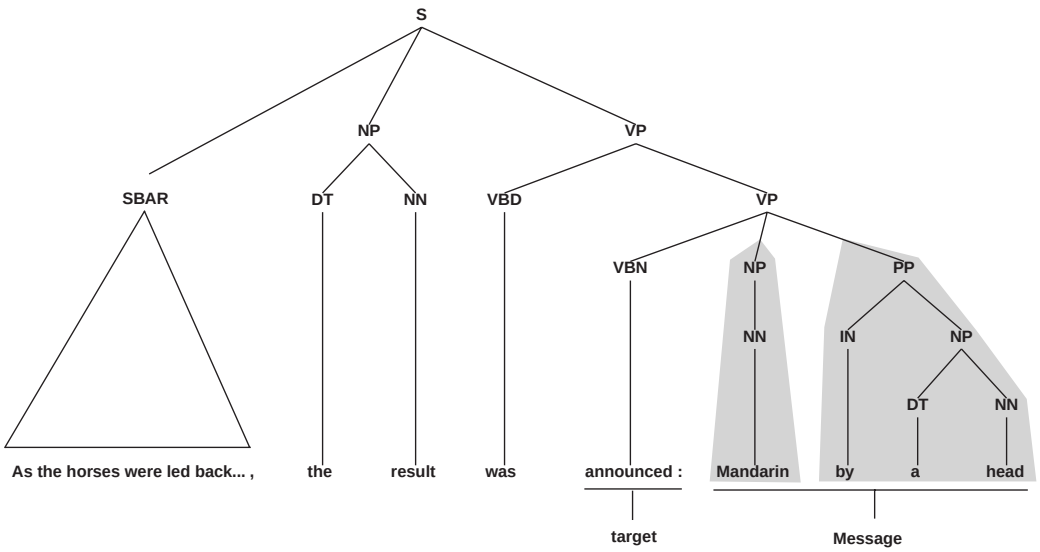

Figure 6: An example of overlap between identified frame elements and the true boundaries: the shaded areas represent frame elements identified by the classifier, with the human annotation below the sentence.

When the automatically identified constituents were fed through the role labeling system described above, 79.6% of the constituents which had been correctly identified in the first stage were assigned the correct role in the second, roughly equivalent to the performance when assigning roles to constituents identified by hand. A more sophisticated integrated system for identifying and labeling frame elements is described in Section 8.1.

6 Thematic Roles

In order to investigate the degree to which our system is dependent on the set of semantic roles used, we performed experiments using abstract, general semantic roles such as AGENT, PATIENT, and GOAL. Such roles were proposed in theories of linking such as Fillmore (1968) and Jackendoff (1972) to explain the syntactic realization of semantic arguments. This level of roles, often called **thematic roles**, was seen as useful for expressing generalizations such as "If a sentence has an AGENT, the AGENT will occupy the subject position." Such correlations might enable a statistical system to generalize from one semantic domain to another.

Recent work on linguistic theories of linking has attempted to explain syntactic realization in

terms of the fundamentals of verbs' meaning — see Levin and Hovav (1996) for a survey of a number of theories. While such an explanation is desirable, our goal is much more modest: an automatic procedure for identifying semantic roles in text, and we aim to use abstract roles as a means of generalizing from limited training data in various semantic domains. We see this effort as consistent with various theoretical accounts of the underlying mechanisms of argument linking, since the various theories all admit some sort of generalization between the roles of specific predicates.

To this end, we developed a correspondence from frame-specific roles to a set of abstract thematic roles. Since there is no canonical set of abstract semantic roles, we decided upon the list shown in Table 8. We are interested in adjuncts as well as arguments, leading to roles such as DEGREE not found in many theories of verb-argument linking. The difficulty of fitting many relations into standard categories such as AGENT and PATIENT led us to include other roles such as TOPIC. In all, we used 18 roles, a somewhat richer set than often used, but still much more restricted than the frame-specific roles. Even with this enriched set, not all frame-specific roles fit neatly into one category.

Role	*Example*
AGENT	**Henry** *pushed* the door open and went in.
CAUSE	Jeez, **that** *amazes* me as well as riles me.
DEGREE	I **rather** *deplore* the recent manifestation of Pop; it doesn't seem to me to have the intellectual force of the art of the Sixties.
EXPERIENCER	It may even have been that **John** *anticipating* his imminent doom ratified some such arrangement perhaps in the ceremony at the Jordan.
FORCE	If this is the case can it be *substantiated* **by evidence from the history of developed societies**?
GOAL	Distant across the river the towers of the castle rose against the sky straddling the only land *approach* **into Shrewsbury**.
INSTRUMENT	In the children with colonic contractions **fasting motility** did not *differentiate* children with and without constipation.
LOCATION	These fleshy appendages are used to detect and *taste* food **amongst the weed and debris on the bottom of a river**.
MANNER	His brow *arched* **delicately**.
NULL	Yet while she had no intention of surrendering her home, **it** would be *foolish* to let the atmosphere between them become too acrimonious.
PATIENT	As soon as a character lays a hand on this item, the skeletal Cleric *grips* **it** more tightly.
PATH	The dung-collector *ambled* slowly **over**, one eye on Sir John.
PERCEPT	What is *apparent* is **that this manual is aimed at the non-specialist technician, possibly an embalmer who has good knowledge of some medical procedures**.
PROPOSITION	It says that rotation of partners does not *demonstrate* **independence**.
RESULT	All the arrangements for stay-behind agents in north-west Europe collapsed, but Dansey was able to *charm* most of the governments in exile in London **into recruiting spies**.
SOURCE	He heard the sound of liquid slurping in a metal container as Farrell *approached* him **from behind**.
STATE	Rex *spied* out Sam Maggott **hollering at all and sundry and making good use of his over-sized red gingham handkerchief**.
TOPIC	He said, "We would urge people to be aware and be *alert* **with fireworks** because your fun might be someone else's tragedy."

Table 8: Abstract Semantic Roles, with representative examples from the FrameNet corpus

An experiment was performed replacing each role tag in the training and test data with the corresponding thematic role, and training the system as described above on the new dataset. Results

were roughly comparable for the two types of semantic roles: overall performance was 82.1% for thematic roles, compared to 80.4% for frame-specific roles. This reflects the fact that most frames had a one-to-one mapping from frame-specific to abstract roles, so the tasks were largely equivalent. We expect abstract roles to be most useful when generalizing to predicates and frames not found in the training data, a future goal of our research.

One interesting consequence of using abstract roles is that they allow us to more easily compare the system's performance on different roles because of the smaller number of categories. This breakdown is shown in Table 9. Results are given for two systems: the first assumes that the frame element boundaries are known and and the second finds them automatically. The second system, which is described in Section 8.1, corresponds to the righthand two columns in Table 9. The labeled recall shows how often the frame element is correctly identified, while the unlabeled recall column show how often a constituent with the given role is correctly identified as being a frame element, even if it is incorrectly labeled as a different frame element.

EXPERIENCER and AGENT are the roles that are correctly identified most often — two similar roles generally found as the subject for complementary sets of verbs. The unlabeled recall column shows that these roles are easy to find in the sentence, as a predicate's subject is almost always a frame element, and the known boundaries column shows that they are also not often confused with other roles when it is known that they are frame elements. The two most difficult roles in terms of unlabeled recall, MANNER and DEGREE, are typically realized by adverbs or prepositional phrases and considered adjuncts. It is interesting to note that these are considered in FrameNet to be *general* frame elements that can be used in any frame.

		known boundaries	unknown boundaries	
Role	Number	% correct	labeled recall	unlabeled recall
Agent	2401	92.8	76.7	80.67
Experiencer	333	91.0	78.7	83.48
Source	503	87.3	67.4	74.16
Proposition	186	86.6	56.5	64.52
State	71	85.9	53.5	61.97
Patient	1161	83.3	63.1	69.08
Topic	244	82.4	64.3	72.13
Goal	694	82.1	60.2	69.60
Cause	424	76.2	61.6	73.82
Path	637	75.0	63.1	63.42
Manner	494	70.4	48.6	59.72
Percept	103	68.0	51.5	65.05
Degree	61	67.2	50.8	60.66
Null	55	65.5	70.9	85.45
Result	40	65.0	55.0	70.00
Location	275	63.3	47.6	63.64
Force	49	59.2	40.8	63.27
Instrument	30	43.3	30.0	73.33
(other)	406	57.9	40.9	63.05
Total	8167	82.1	63.6	72.10

Table 9: Performance broken down by abstract role. The third column represents accuracy where frame element boundary are given to the system, while the fourth and fifth columns reflect finding the boundaries automatically. Unlabeled recall includes cases that were identified as a frame element but given the wrong role.

7 Generalizing Lexical Statistics

As can be seen from Table 2, information about the head word of a constituent is valuable in predicting the constituent's role. Of all the distributions presented, $P(r|h, pt, t)$ predicts the correct role most often (87.4% of the time) when training data for a particular head word has been seen. However, due to the large vocabulary of possible head words, it also has the smallest *coverage*, meaning it is likely that for a given case in the test data, no frame element with the same head word will have been seen in the set of training sentences for the target word in question. To capitalize on the information provided by the head word, we wish to find a way to generalize from head words seen in the training data to other head words. In this section we compare three different approaches to the task of generalizing over head words: **automatic clustering** of a large vocabulary of head words to identify words with similar semantics, use of a hand-built **ontological resource**, WordNet, to organize head words in a semantic hierarchy, and **bootstrapping** to make use of unlabeled data in training the system. We will focus on frame elements filled by noun phrases, which comprise roughly half the total.

7.1 Automatic Clustering

In order to find groups of nouns with similar semantic properties, an automatic clustering was performed using the general technique of Lin (1998). This technique is based on the expectation that words with similar semantics will tend to co-occur with the same other sets of words. For example, nouns describing foods will tend to occur as direct objects of verbs such "eat" as well as "devour", "savor", etc. The algorithm below attempts to find such patterns of co-occurrence from the counts of grammatical relations between specific words in the corpus, without the use of any external knowledge or semantic representation.

We extracted verb-direct object relations from an automatically parsed version of the British National Corpus, using the parser of Carroll and Rooth (1998). [3] Clustering was performed using the probabilistic co-occurrence model of Hofmann and Puzicha (1998). [4] According to this model, the two observed variables, in this case the verb and the head noun of its object, can be considered independent given the value of a hidden cluster variable, c:

$$P(n, v) = \sum_c P(c)P(n|c)P(v|c)$$

One begins by setting *a priori* the number of values that c can take, and using the EM algorithm to estimate the distributions $P(c)$, $P(n|c)$ and $P(v|c)$. Deterministic annealing was used in order to prevent overfitting of the training data.

We are interested only in the clusters of nouns given by the distribution $P(n|c)$ — the verbs and the distribution $P(v|c)$ are thrown away once training is complete. Other grammatical relations besides direct object could be used, as could a set of relations. We used the direct object (following other clustering work such as Pereira, Tishby, and Lee (1993)) because it is particularly likely to exhibit semantically significant selectional restrictions.

A total of 2,610,946 verb-object pairs were used as training data for the clustering, with a further 290,105 pairs used as a cross-validation set to control the parameters of the clustering algorithm. Direct objects were identified as noun phrases directly under a verb phrase node — not a perfect technique, since it also finds nominal adjuncts such as "I start *today*". Forms of the verb "to be" were excluded from the data, as its co-occurrence patterns are not semantically informative. The number of values possible for the latent cluster variable was set to 256. (Comparable results were

[3] We are indebted to Mats Rooth for providing us with the parsed corpus.

[4] For other NLP applications of the probabilistic clustering algorithm, see e.g. (Rooth et al., 1999). For application to language modeling, see (Gildea and Hofmann, 1999).

found with 64 clusters; the use of deterministic annealing prevents a large numbers of clusters from resulting in overfitting.)

The soft clustering of nouns thus generated is used as follows: for each example in the frame-element-annotated training data, probabilities for values of the hidden cluster variable were calculated using Bayes' rule:

$$P(c|h) = \frac{P(h|c)P(c)}{\sum_i P(h|c_i)P(c_i)}$$

The clustering was applied only to noun phrase constituents; the distribution $P(n|c)$ from the clustering is used as a distribution $P(h|c)$ over noun head words.

Using the cluster probabilities, a new estimate of $P(r|c, nt, t)$ is calculated for cases where nt, the nonterminal or syntactic category of the constituent, is NP:

$$P(r|c, nt, t) = \frac{\sum_{j:nt_j=nt, t_j=t, r_j=r} P(c_j|h_j)}{\sum_{j:nt_j=nt, t_j=t} P(c_j|h_j)}$$

During testing, a smoothed estimate of $P(r|h, nt, t)$ is calculated as $\sum_c P(r|c, nt, t)P(c|h)$, again using $P(c|h) = \frac{P(h|c)P(c)}{\sum_i P(h|c_i)P(c_i)}$.

As with the other methods of generalization described in this section, automatic clustering was applied only to noun phrases, which represent 50% of the constituents in the test data. We would not expect head word to be as valuable for other syntactic types. The second most common category is prepositional phrases. The head of a prepositional phrase is considered to be the preposition according to the Collins/Magerman rules we use, and because the set of prepositions is small, coverage is not as great of a problem. Furthermore, the preposition is often a direct indicator of the semantic role. (A more complete model might distinguish between cases where the preposition serves as a case or role marker, and others where it is semantically informative, with clustering being performed on the preposition's object in the former case. We did not attempt to make this distinction.)

Distribution	*Coverage*	*Accuracy*	*Performance*
$P(r\|h, pt, t)$	41.6	87.0	36.1
$\sum_c P(r\|c, pt, t)P(c\|h)$	97.9	79.7	78.0
Interpolation of unclustered distributions	100.0	83.4	83.4
Unclustered distributions + clustering	100.0	85.0	85.0

Table 10: Clustering results on NP constuents only: 4086 instances.

Table 10 shows results for the use of automatic clustering on constituents identified by the parser as noun phrases. As can be seen, the vocabulary used for clustering includes almost all (97.9%) of the test data, and the decrease in accuracy from direct lexical statistics to clustered statistics is relatively small (from 87.0% to 79.7%). When combined with the full system described above, clustered statistics increase performance on NP constituents from 83.4% to 85.0% (statistically significant at $p = .05$). Over the entire test set, this translates into an improvement from 80.4% to 81.2%.

7.2 Using a Semantic Hierarchy: Wordnet

The automatic clustering described above can be seen as an imperfect method of deriving semantic classes from the vocabulary, and we might expect a hand-developed set of classes to do better. We tested this hypothesis using Wordnet (Fellbaum, 1998), a freely available semantic hierarchy. The basic technique, when presented with a head word for which no training examples had been seen,

Distribution	*Coverage*	*Accuracy*	*Performance*
$P(r\|h, pt, t)$	41.6	87.0	36.1
$Wordnet : P(r\|s, pt, t)$	80.8	79.5	64.1
Interpolation of unclustered distributions	100.0	83.4	83.4
Unclustered distributions + Wordnet	100.0	84.3	84.3

Table 11: Wordnet results on NP constituents only: 4086 instances.

was to ascend type hierarchy until reaching a level for which training data are available. To do this, counts of training data were percolated up the semantic hierarchy in a technique similar to that of, for example, McCarthy (2000). For each training example, the count $\#(r, s, pt, t)$ was incremented in a table indexed by the semantic role r, Wordnet sense s, phrase type pt, and target word t, for each Wordnet sense s above the head word h in the hypernym hierarchy. In fact, the Wordnet hierarchy is not a tree, but rather includes multiple inheritance. For example, "person" has as hypernyms both "life form" and "causal agent". In such cases, we simply took the first hypernym listed, effectively converting the structure into a tree. A further complication is that several Wordnet senses are possible for a given head word. We simply used the first sense listed for each word – a word sense disambiguation module capable of distinguishing Wordnet senses might improve our results.

As with the clustering experiments reported above, the Wordnet hierarchy was used only for noun phrases. The Wordnet hierarchy does not include pronouns — in order to increase coverage, the words "I", "me", "you", "he", "she", "him", "her", "we", and "us" were added as hyponyms of "person". Pronouns which refer to inanimate, or both animate and inanimate, objects, were not included. In addition, the CELEX English lexical database (Celex, 1993) was used to convert plural nouns to their singular forms.

As can been seen from the results in Table 11, accuracy for the Wordnet technique is roughly the same as the automatic clustering results in Table 10 — 84.3% on NPs, as opposed to 85.0% with automatic clustering. This indicates that the error introduced by the unsupervised clustering is roughly equivalent to the error caused by our arbitrary choice of the first Wordnet sense for each word and the first hypernym for each Wordnet sense. However, coverage for the Wordnet technique is lower, largely due to the absence of proper nouns from Wordnet, as well as the absence of non-animate pronouns (in which we include both personal pronouns such as "it" and "they" and indefinite pronouns such as "something" and "anyone"). A proper nouns dictionary would be likely to help improve coverage, and a module for anaphora resolution might help cases with pronouns, with or without the use of Wordnet. The conversion of plural forms to singular base forms was an important part of the success of the Wordnet system, increasing coverage from 71.0% to 80.8%. Of the remaining 19.2% of all noun phrases not covered by the combination of lexical and Wordnet sense statistics, 22% consisted of head words defined in Wordnet, but for which no training data were available for any hypernym, and 78% consisted of head words not defined in Wordnet.

7.3 Bootstrapping from Unannotated Data

A third way of attempting to improve coverage of the lexical statistics is to "bootstrap", or label unannotated data with the automatic system, and use the (imperfect) result as further training data. This can be considered a variant of the EM algorithm, although we use the single most likely hypothesis for the unannotated data, rather than calculating the expectation over all hypotheses. Only one iteration of training on the unannotated data was performed.

The unannotated data used consisted of 156,590 sentences containing the target words under investigation, increasing the total amount of data available to roughly six times the 36,995 annotated training sentences.

Table 12 shows results on noun phrases for the bootstrapping method. The accuracy of a system trained only on data from the automatic labeling (P_{auto}) is 81.0%, reasonably close to the 87.0% for

Distribution	*Coverage*	*Accuracy*	*Performance*
$P_{train}(r\|h,pt,t)$	41.6	87.0	36.1
$P_{auto}(r\|h,pt,t)$	48.2	81.0	39.0
$P_{train+auto}(r\|h,pt,t)$	54.7	81.4	44.5
P_{train}, backoff to P_{auto}	54.7	81.7	44.7
Interpolation of unclustered distributions	100	83.4	83.4
Unclustered distributions + P_{auto}	100	83.2	83.2

Table 12: Bootstrapping results on NP constituents only: 4086 instances.

the system trained only on annotated data (P_{train}). Combining the annotated and automatically labeled data increases coverage from 41.6% to 54.7%, and performance to 44.5%. Because the automatically labeled data are not as accurate as the annotated data, we can do slightly better by using the automatic data only in cases where no training data is available, "backing off" to the distribution P_{auto} from P_{train}. The last row of Table 12 shows results with P_{auto} incorporated into the backoff lattice of all the features of Figure 3, which actually resulted in a slight decrease in performance from the system without the bootstrapped data, shown in the second to last row. This is presumably because, although the system trained on automatically labeled data performed with reasonable accuracy, many of the cases it classifies correctly overlap with the training data. In fact our backing-off estimate of $P(r|h,pt,t)$ only classifies correctly 66% of the additional cases that it covers over $P_{train}(r|h,pt,t)$.

7.4 Discussion

The three methods of generalizing lexical statistics each had roughly equivalent accuracy on cases for which they were able to come up with an estimate of the role probabilities for unseen head words. The differences between the three were primarily due to how much they could improve the *coverage* of the estimator, that is, how many new noun heads they were able to handle. The automatic clustering method performed by far the best on this metric; only 2.1% of test cases were unseen in the data used for calculating the clustering. This indicates how much can be achieved with unsupervised methods given very large training corpora. The bootstrapping technique described here, while having a similar unsupervised flavor, made use of much less data than the corpus used for noun clustering. Unlike the probabilistic clustering, the bootstrapping technique can only make use of sentences containing the target words in question. The Wordnet experiment, on the other hand, indicates both the usefulness of hand-built resources when they apply and the difficulty of attaining broad coverage with such resources. We plan to combine the three systems described to test whether the gains are complementary or overlapping.

8 Verb Argument Structure

One of the primary difficulties in labeling semantic role is that one predicate may be used with different argument structures: for example in the sentences "He opened the door" and "The door opened", the verb "open" assigns different semantic roles to its syntactic subject. In this section we compare two strategies for handling this type of alternation: a sentence-level feature for frame element groups, and a subcategorization feature for the syntactic uses of verbs.

8.1 Priors on Frame Element Groups

The system described above for classifying frame elements makes an important simplifying assumption: it classifies each frame element independently of the decisions made for the other frame elements

Frame Element Group	Example Sentences
{ EVALUEE }	Holman would characterise this as **blaming** [$_{Evaluee}$ the poor] .
{ JUDGE, EVALUEE, REASON }	The letter quotes Black as saying that [$_{Judge}$ white and Navajo ranchers] misrepresent their livestock losses and **blame** [$_{Reason}$ everything] [$_{Evaluee}$ on coyotes] . [$_{Judge}$ She] **blames** [$_{Evaluee}$ the Government] [$_{Reason}$ for failing to do enough to help] .
{ JUDGE, EVALUEE }	The only dish she made that we could tolerate was [$_{Evaluee}$ syrup tart which] [$_{Judge}$ we] **praised** extravagantly with the result that it became our unhealthy staple diet.

Table 13: Sample frame element groups for the verb "blame".

Frame Element Group	*Prob.*
{ EVAL, JUDGE, REAS }	0.549
{ EVAL, JUDGE }	0.160
{ EVAL, REAS }	0.167
{ EVAL }	0.097
{ EVAL, JUDGE, ROLE }	0.014
{ JUDGE }	0.007
{ JUDGE, REAS }	0.007

Table 14: Frame element groups for the verb "blame" in the JUDGMENT frame.

in the sentence. In this section we relax this assumption, and present a system which can make use of the information that, for example, a given target word requires that one role always be present, or that having two instances of the same role is extremely unlikely.

In order to capture this information, we introduce the notion of a **frame element group**, which is the set of frame element roles present in a particular sentence (technically a multiset, as duplicates are possible, though quite rare). Frame element groups, or FEGs, are *unordered* — examples are shown in Table 13. [5]

Our system for choosing the most likely overall assignment of roles for all the frame elements of a sentences uses an approximation which we derived beginning with the true probability of the optimal role assignment r^*:

$$r^* = argmax_{r_{1..n}} P(r_{1..n}|t, f_{1..n})$$

where $P(r_{1..n}|t, f_{1..n})$ represents the probability of an overall assignment of roles r_i to each of the n constituents of a sentence, given the target word t and the various features f_i of each of the constituents. In the first step we apply Bayes' rule to this quantity, and in the second we make the assumption that the features of the various constituents of a sentence are independent given the target word and each constituent's role:

$$r^* = argmax_{r_{1..n}} P(r_{1..n}|t) \frac{P(f_{1..n}|r_{1..n}, t)}{P(f_{1..n}|t)}$$

[5] The FrameNet corpus recognized three types of "null instantiated" frame elements (Fillmore, 1986), which are implied but do not appear in the sentence. An example of null instantiation is the sentence "Have you eaten?", where "food" is understood. We did not attempt to identify such null elements, and any null instantiated roles are not included in the sentence's frame element group. This increases the variability of observed FEGs, as a predicate may require a certain role but allow it to be null instantiated.

We estimate the prior over frame element assignments as the probability of the frame element groups, represented with the set operator {}:

$$r^* = argmax_{r_{1..n}} P(\{r_{1..n}\}|t) \prod_i P(f_i|r_i, t)$$

We then apply Bayes' rule again:

$$r^* = argmax_{r_{1..n}} P(\{r_{1..n}\}|t) \prod_i \frac{P(r_i|f_i, t)P(f_i|t)}{P(r_i|t)}$$

and finally discard the feature prior $P(f_i|t)$ as being constant over the $argmax$ expression:

$$r^* = argmax_{r_{1..n}} P(\{r_{1..n}\}|t) \prod_i \frac{P(r_i|f_i, t)}{P(r_i|t)}$$

This leaves us with an expression in terms of the prior for frame element groups of a particular target word $P(\{r_{1..n}\}|t)$, the local probability of a frame element given a constituent's features $P(r_i|f_i, t)$ on which our previous system was based, and the individual priors for the frame elements chosen $P(r_i|t)$. This formulation can be used either to assign roles where the frame element boundaries are known, or where they are not, as we will discuss later in this section.

Calculating empirical FEG priors from the training data is relatively straightforward, but the sparseness of the data presents a problem. In fact, 15% of the test sentences had an FEG not seen in the training data for the target word in question. Using the empirical value for the FEG prior, these sentences could never be correctly classified. For this reason, we introduce a smoothed estimate of the FEG prior, consisting of a linear interpolation of the empirical FEG prior and the product, for each each possible frame element, of the probability of being present or not present in a sentence given the target word:

$$\lambda P(\{r_{1..n}\}|t) + (1 - \lambda) \left[\prod_{r \in r_{1..n}} P(r \in FEG|t) \prod_{r \notin r_{1..n}} P(r \notin FEG|t) \right]$$

The value of λ was empirically set to maximize performance on the development set; a value of 0.6 yielded performance of 81.6%, a significant improvement over the 80.4% of the baseline system. Results were relatively insensitive to the exact value of λ.

Up to this point, we have considered separately the problems of labeling roles given that we know where the boundaries of the frame elements lie (Section 4, as well as Section 7) and finding the constituents to label in the sentence (Section 5). We know turn to combining the two systems described above into a complete role labeling system. We use equation 11 to estimate the probability that a constituent is a frame element, repeated below:

$$P(fe|p, h, t) = \lambda_1 P(fe|p) + \lambda_2 P(fe|p, t)) + \lambda_3 P(fe|h, t)$$

where p is the *path* through the parse tree from the target word to the constituent, t is the target word, and h is the constituent's head word.

The first two rows of Table 15 show the results obtained by deciding which constituents are frame elements by setting the threshold on the probability $P(fe|p, h, t)$ to 0.5, and then running the labeling system of Section 4 on the resulting set of constituent. The first two columns of results show precision and recall for the task of identifying frame element boundaries correctly. The second pair of columns gives precision and recall for the combined task of boundary identification and role labeling — to be counted as correct, the frame element must have the correct boundary and subsequently be labeled with the correct role.

Method	FE Prec.	FE Recall	Labeled Prec.	Labeled Recall
Boundary id. + baseline role labeler	72.6	63.1	67.0	46.8
Boundary id. + labeler w/ FEG priors	72.6	63.1	65.9	46.2
Integrated boundary id. and labeling	74.0	70.1	64.6	61.2

Table 15: Combined results on boundary identification and role labeling.

Contrary to our results using human annotated boundaries, incorporating FEG priors into the system had a negative effect. No doubt this is due to introducing a dependency on other frame element decisions which may be incorrect — the use of FEG priors causes errors in boundary identification to be compounded.

One way around this problem is to integrate boundary identification with role labeling, allowing the FEG priors and the role labeling decisions to have an effect on decisions as to which constituents are frame elements. This was accomplished by extending the formulation

$$argmax_{r_{1..n}} P(\{r_{1..n}\}|t) \prod_i \frac{P(r_i|f_i,t)}{P(r_i|t)}$$

to include FE identification decisions:

$$argmax_{r_{1..n}} P(\{r_{1..n}\}|t) \prod_i \frac{P(r_i|f_i,fe_i,t)P(fe_i|f_i)}{P(r_i|t)}$$

where fe_i is a binary variable indicating that a constituent is a frame element and $P(fe_i|f_i)$ is calculated as above. When fe_i is true, role probabilities are calculated as before; when fe_i is false, r_i assumes an empty role with probability one, and is not included in the Frame Element Group represented by $\{r_{1..n}\}$.

One caveat using this integrated approach is its exponential complexity: each combination of role assignments to constituents is considered, and the number of combinations is exponential in the number of constituents. While this did not pose a problem when only the annotated frame elements were under consideration, now we must include every parse constituent with a non-zero probability for $P(fe_i|f_i)$. In order to make the computation tractable, we implement a pruning scheme: hypotheses are extended by choosing assignments for one constituent at a time, and only the top m hypotheses are retained for extension by assignments to the next constituent. Here we set $m = 10$ after experimentation showed that increasing m yielded no significant improvement.

Results for the integrated approach are shown in the last row of Table 15. Allowing role assignments to influence boundary identification improves results on both the unlabeled boundary identification task, and the combined identification and labeling task. The integrated approach puts us in a different portion of the precision/recall curve from the results in the first two rows, as it returns a higher number of frame elements (7736 vs. 5719). A more direct comparison can be made by lowering the probability threshold for frame element identification from .5 to .35, in order to force the non-integrated system to return the same number of frame elements as the integrated system. This yields a frame element identification precision of 71.3% and recall of 67.6%, and labeled precision of 60.8% and recall of 57.6%, which is dominated by the result for the integrated system. The integrated system does not have a probability threshold to set; nonetheless it comes closer to identifying the correct number of frame elements (8167) than does the independent boundary identifier when the theoretically optimal threshold of .5 is used with the latter.

8.2 Subcategorization

Recall that use of the FEG prior was motivated by the ability of verbs to assign differing roles to the same syntactic position. For example, the verb "open" assigns different roles to the syntactic

subject in "He opened the door" and "The door opened". In this section we consider a different feature motivated by these problems: the syntactic subcategoriation of the verb. For example, the verb "open" seems to be more likely to assign the role PATIENT to its subject in an intransitive context, and AGENT to its subject in a transitive context. Our use of a subcategorization feature was intended to differentiate between transitive and intransitive uses of a verb.

The feature used was the identity of the phrase structure rule expanding the target word's parent node in the parse tree, as shown in Figure 7. For example, for "He closed the door", with "close" as the target word, the subcategorization feature would be "VP → V NP". The subcategorization feature was used only when the target word was a verb, and the various part of speech tags for verb forms were collapsed. It is important to note that we are not able to distinguish complements from adjuncts, and our subcategorization feature could be sabotaged by cases such as "The door closed yesterday", as in the Penn Treebank style, "yesterday" is considered an NP with tree structure equivalent to that of a direct object. Our subcategorization feature is fairly specific, as for example the addition of an ADVP to a verb phrase will result in a different value. We tested variations of the feature that counted the number of NPs in a VP or the total number of children of the VP, with no significant change in results.

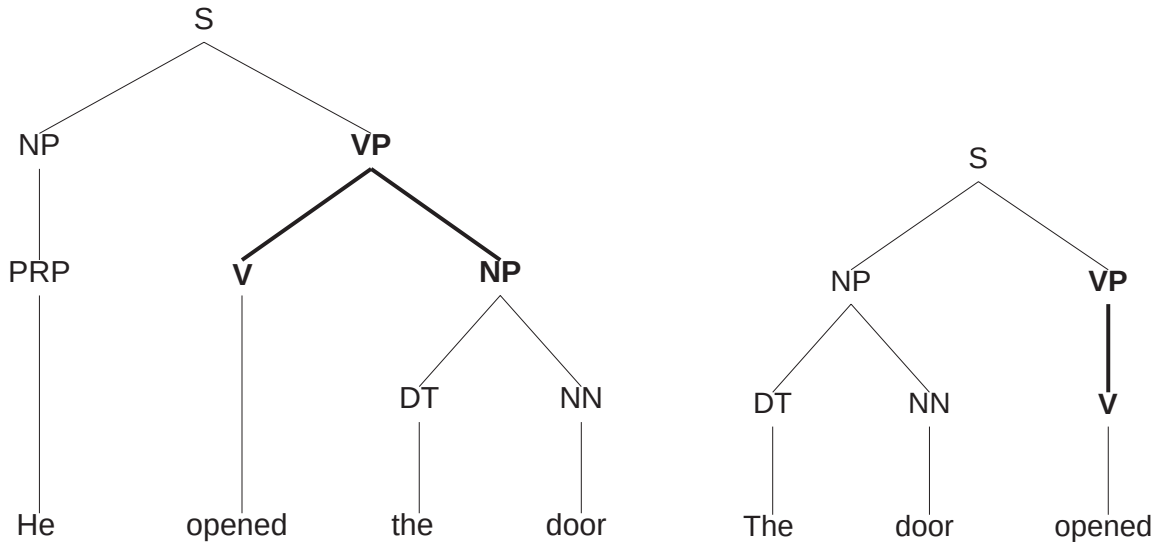

Figure 7: Two subcategorizations for the target word "open". The relevant production in the parse tree is highlighted. On the left, the value of the feature is "VP → V NP"; on the right it is "VP → V".

The subcategorization feature was used in conjunction with the *path* feature, which represents the sequence of non-terminals along the path through the parse tree from the target word to the constituent representing a frame element. Making use of the new subcategorization feature by adding the distribution $P(r|subcat, path, t)$ to the lattice of distributions in the baseline system resulted in a slight improvement to 80.8% performance from 80.4%. As with the *gf* feature in the baseline system, it was found beneficial to use the *subcat* feature only for NP constituents.

8.3 Discussion

Combining the Frame Element Group priors and subcategorization feature into a single system resulted in performance of 81.6%, no improvement over using FEG priors without subcategorization. We suspect that the two seemingly different approaches in fact provide similar information. For

example, in our hypothetical example of the sentences "He closed the door" vs. "The door closed", the verb "close" would have high priors for the FEGs { AGENT, THEME } and { THEME }, but a low prior for { AGENT }. In sentences with only one candidate frame element (the subject in "The door closed"), the use of the FEG prior will cause it to be labeled THEME even when the feature probabilities prefer labeling a subject as AGENT. Thus the FEG prior, by representing the set of arguments the predicate is likely to take, essentially already performs the functions of the subcategorization feature.

The FEG prior allows us to introduce a dependency between the classifications of the sentence's various constituents with a single parameter. Thus, it can handle the alternation of our example without, for example, introducing the role chosen for one constituent as an additional feature in the probability distribution for the next constituent's role. It appears that because introducing additional features can further fragment our already sparse data, it is preferable to have a single parameter for the FEG prior.

An interesting result reinforcing this conclusion is that some of the argument structure-related features which aided the system when individual frame elements were considered independently are unnecessary when using FEG priors. Removing the features *passive* and *position* from the system and using a smaller lattice of only the distributions not using these features yields performance of 82.8% on the role labeling task using hand-annotated boundaries. We believe that because these features pertain to syntactic alternations in how arguments are realized, they overlap with the function of the FEG prior. Adding unnecessary features to the system can reduce performance by fragmenting the training data.

9 Conclusion

Our preliminary system is able to automatically label semantic roles with fairly high accuracy, indicating promise for applications in various natural language tasks. Lexical statistics computed on constituent head words were found to be the most important of the features used. While lexical statistics are quite accurate on the data covered by observations in the training set, the sparsity of the data when conditioned on lexical items meant that combining features was the key to high overall performance. While the combined system was far more accurate than any feature taken alone, the specific method of combination used was less important. Various methods of extending the coverage of lexical statistics indicated that the broader coverage of an automatic clustering outweighed its imprecision. Including priors over sets of frame elements in a sentence was found to be an effective way of modeling dependencies between individual classification decision without adding too much complexity to the system.

We plan to continue this work by integrating semantic role identification with parsing, by bootstrapping the system on larger, and more representative, amounts of data, and by attempting to generalize from the set of predicates chosen by FrameNet for annotation to general text. One strategy for this last goal would the combination of FrameNet data with named entity systems for recognizing times, dates, and locations — allowing the effort which has gone into recognizing these items, typically adjuncts, with the FrameNet data, which is more focused on arguments. Another avenue for expanding the system is some type of clustering of the target predicates and frames, which are currently considered independently.

References

Baker, Collin F., Charles J. Fillmore, and John B. Lowe. 1998. The Berkeley FrameNet project. In *Proceedings of the COLING-ACL*, Montreal, Canada.

Blaheta, Don and Eugene Charniak. 2000. Assigning function tags to parsed text. In *Proceedings of the 1st Annual Meeting of the North American Chapter of the ACL (NAACL)*, Seattle, Washington.

Carroll, Glenn and Mats Rooth. 1998. Valence induction with a head-lexicalized PCFG. In *Proceedings of the 3rd Conference on Empirical Methods in Natural Language Processing (EMNLP 3)*, Granada, Spain.

Celex. 1993. The CELEX lexical database. Centre for Lexical Information, Max Planck Institute for Psycholinguistics.

Collins, Michael. 1997. Three generative, lexicalised models for statistical parsing. In *Proceedings of the 35th Annual Meeting of the ACL.*

Dowty, David R. 1991. Thematic proto-roles and argument selction. *Language*, 67(3):547–619.

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

Fillmore, Charles J. 1968. The case for case. In Emmon W. Bach and Robert T. Harms, editors, *Universals in Linguistic Theory.* Holt, Rinehart & Winston, New York, pages 1–88.

Fillmore, Charles J. 1971. Some problems for case grammar. In R. J. O'Brien, editor, *22nd annual Round Table. Linguistics: developements of the sixties – viewpoints of the seventies*, volume 24 of *Monograph Series on Language and Linguistics.* Georgetown University Press, Washington D.C., pages 35–56.

Fillmore, Charles J. 1976. Frame semantics and the nature of language. In *Annals of the New York Academy of Sciences: Conference on the Origin and Development of Language and Speech*, volume 280, pages 20–32.

Fillmore, Charles J. 1986. Pragmatically controlled zero anaphora. In *Proceedings of the 12th Annual Meeting of the Berkeley Linguistics Society*, pages 95–107, Berkeley, CA.

Fillmore, Charles J. and Collin F. Baker. 2000. FrameNet: Frame semantics meets the corpus. In *Linguistic Society of America*, January.

Gildea, Daniel and Thomas Hofmann. 1999. Probabilistic topic analysis for language modeling. In *Eurospeech-99*, Budapest.

Hearst, Marti. 1999. Untangling text data mining. In *Proceedings of the 37rd Annual Meeting of the ACL.*

Hofmann, Thomas and Jan Puzicha. 1998. Statistical models for co-occurrence data. Memo, Massachussetts Institute of Technology Artificial Intelligence Laboratory, February.

Jackendoff, Ray. 1972. *Semantic Interpretation in Generative Grammar.* MIT Press, Cambridge, Massachusetts.

Lapata, Maria and Chris Brew. 1999. Using subcategorization to resolve verb class ambiguity. In *Joint SIGDAT Conference on Empirical Methods in NLP and Very Large Corpora*, Maryland.

Levin, Beth and Malka Rappaport Hovav. 1996. From lexical semantics to argument realization. manuscript.

Lin, Dekang. 1998. Automatic retrieval and clustering of similar words. In *Proceedings of the COLING-ACL*, Montreal, Canada.

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

McCarthy, Diana. 2000. Using semantic preferences to identify verbal participation in role switching alternations. In *Proceedings of the 1st Annual Meeting of the North American Chapter of the ACL (NAACL)*, Seattle, Washington.

Miller, Scott, David Stallard, Robert Bobrow, and Richard Schwartz. 1996. A fully statistical approach to natural language interfaces. In *Proceedings of the 34th Annual Meeting of the ACL.*

Pereira, Fernando, Naftali Tishby, and Lilian Lee. 1993. Distributional clustering of English words. In *Proceedings of the 31st Annual Meeting of the ACL*, Columbus, Ohio.

Pollard, Carl and Ivan A. Sag. 1994. *Head-Driven Phrase Structure Grammar.* University of Chicago Press, Chicago.

Riloff, Ellen. 1993. Automatically constructing a dictionary for information extraction tasks. In *Proceedings of the Eleventh National Conference on Artificial Intelligence (AAAI).*

Riloff, Ellen and Mark Schmelzenbach. 1998. An empirical approach to conceptual case frame acquisition. In *Proceedings of the Sixth Workshop on Very Large Corpora*.

Rooth, Mats, Stefan Riezler, Detlef Prescher, Glenn Carroll, and Franz Beil. 1999. Inducing a semantically annotated lexicon via EM-based clustering. In *Proceedings of the 37th Annual Meeting of the ACL*, College Park, Maryland.

Schank, R. C. 1972. Conceptual dependency: a theory of natural language understanding. *Cognitive Psychology*, 3:552–631.

Somers, H. L. 1987. *Valency and Case in Computational Linguistics*. Edinburgh University Press, Edinburgh, Scotland.

Van Valin, Robert D. 1993. A synopsis of role and reference grammar. In Robert D. Van Valin, editor, *Advances in Role and Reference Grammar*. John Benjamins Publishing Company, Amsterdam, pages 1–166.

Winograd, Terry. 1972. Understanding natural language. *Cognitive Psychology*, 3(1). Reprinted as a book by Academic Press, 1972.