A Generative Model for Semantic Role Labeling

Cynthia A. Thompson[1], Roger Levy[2], and Christopher D. Manning[2]

[1] School of Computing, University of Utah
Salt Lake City, UT 84112
{cindi}@cs.utah.edu
http://www.cs.utah.edu/~cindi
[2] Departments of Linguistics and Computer Science
Stanford University
Stanford, CA 94305 {rog|manning}@stanford.edu

Abstract. Determining the semantic role of sentence constituents is a key task in determining sentence meanings lying behind a veneer of variant syntactic expression. We present a model of natural language generation from semantics using the FrameNet semantic role and frame ontology. We train the model using the FrameNet corpus and apply it to the task of automatic semantic role and frame identification, producing results competitive with previous work (about 70% role labeling accuracy). Unlike previous models used for this task, our model does not assume that the frame of a sentence is known, and is able to identify *null-instantiated roles*, which commonly occur in our corpus and whose identification is crucial to natural language interpretation.

1 Introduction

A central goal of natural language processing is domain-independent understanding. A useful step towards that goal is the assignment of semantic roles to the (syntactic) constituents of a sentence. Having semantic roles allows one to recognize semantic arguments of a situation, even when expressed in different syntactic configurations. For example the role of an *instrument*, such as a *hammer*, can be recognized, regardless of whether its expression is as the subject of the sentence (*the hammer broke the vase*) or via a prepositional phrase headed by *with*. This paper attempts the task of learning to automatically assign such roles. Identifying such roles and the relationships between them can in turn serve as support for inference about a sentence's meaning, for antecedent resolution, or for other understanding or parsing tasks such as prepositional phrase attachment or word sense disambiguation.

This paper develops a generative model from which one can infer role labels, given sentence constituents and a word from that sentence that is a *predicator*, which takes semantic role arguments. We learn the parameters for this model from a body of examples provided by the FrameNet corpus [1]. The problem and some elements of our approach are similar to that of [2], but the work differs by use of a generative, not a discriminative, model, and by assuming less known information for making the role assignment. A difficulty of this task is that there is limited data available annotated with semantic roles, in comparison to syntactic parsing. As an illustration of this, in the model developed by [2] the most accurate rules only covered 50% of the unseen examples. To overcome the limited amount of training data, we would ultimately like to apply *bootstrapping*, in which limited labeled data are combined with unlabeled data to produce a more accurate model than that trained on unlabeled data alone [3, 4]. Generative models are a natural choice in the case of combining fully and partially annotated data. First, we need to test their capabilities on fully annotated data, such as it exists. This is the focus of the current paper.

Our work can be compared and contrasted with much past work in information extraction [5–7], in which the goal is to extract from text words or phrases that fill a role, such as "acquiring company" or "vehicle," and in which there are often multiple roles of interest. In particular, recent work such as [5] uses Hidden Markov Models, including induction over the structure of the model, for the labeling task. The model we use is similar, but while our goal is also to identify which roles are filled, and identify the words that fill them, we additionally aim to identify the overarching relationship that holds between the roles. We call this relationship the *frame*. Secondly, information extraction normally uses a small number of very domain specific roles, while our corpus has a large number of roles, with many types of roles that apply across domains. The techniques of information extraction may not scale well to large numbers of roles. Also, in information extraction, the labeling task is somewhat tied, semantically, to the domain at hand. These methods also tend to rely on regular structure, such as capitalization or indicator terms drawn from a closed class. Finally, the currently annotated semantic data is primarily at the sentence level, versus entire texts for information extraction.

The acquisition of selectional preferences, or the tendency of verbs to prefer arguments of a particular type, is a second closely related area [8, 9]. In this line of research statistical models are typically trained on parsed sentences to determine verb-subject or verb-direct object relationships. Such information can be useful for prepositional phrase attachment or to help determine the semantic class of a previously unseen word.

In this paper, we show that our generative model for role labeling produces results competitive with previous work in this area. In addition, our model is flexible enough to be used for annotating additional data, thus improving the model and the pool of data available for other researchers. Second, it has the advantage of capturing the case when roles are *null instantiated* in a particular sentence: they are not overtly expressed but their presence is understood implicitly in discourse. While our model handles these roles, we leave to future work a full evaluation of this ability. Finally, it can identify which constituents correspond to role labels of a particular given predicator.

2 Background

In this section we discuss the FrameNet Corpus, the previous work on labeling roles by Gildea and Jurafsky, and the role labeling task in more detail.

2.1 The FrameNet Corpus

FrameNet [1] is a large-scale, domain-independent computational lexicography project organized around the motivating principles of lexical semantics: that systematic correlations can be found between the meaning components of words, principally the semantic roles associated with events, and their combinatorial properties in syntax. This principle has been instantiated at various levels of granularity in different traditions of linguistic research; FrameNet researchers work at an intermediate level of granularity, termed the *frame*. Examples of frames include MOTION_DIRECTIONAL, CONVERSATION, JUDGMENT, and TRANSPORTATION. Frames consist of multiple *lexical units*—a items corresponding to a sense of a word. Examples for the MOTION_DIRECTIONAL frame are *drop* and *plummet*. Also associated with each frame is a set of semantic *roles*. Examples for the MOTION_DIRECTIONAL frame include the moving object, called the THEME; the ultimate destination, the GOAL; the SOURCE; and the PATH.

In addition to frame and role definitions, FrameNet has produced a large number of role-annotated sentences; the sentences are drawn primarily from the British National Corpus. There are two releases of the corpus, FrameNet I and FrameNet II;[3] we present results from both, but have so far focused primarily on the former. For each annotated example sentence, a lexical unit of interest, one which takes arguments, is identified. We will call this word the *predicator*.[4] The words and phrases which participate in the predicator's meaning are labeled with their roles, and the entire sentence is labeled with the relevant frame. Finally, the corpus also includes syntactic category information for each role. We give some examples below, with the frame listed in braces at the beginning, the predicator in bold, and each relevant constituent labeled with its role and phrase type. Note that the last example has a DRIVER role that is null instantiated.

{MOTION_DIRECTIONAL} Mortars lob heavy shells high into the sky so that $[^{\text{NP}}_{\text{THEME}}$ they] **drop** $[^{\text{PP}}_{\text{PATH}}$ down] $[^{\text{PP}}_{\text{GOAL}}$ on the target] $[^{\text{PP}}_{\text{SOURCE}}$ from the sky].

{ARRIVING} He heard the sound of liquid slurping in a metal container as $[^{\text{NP}}_{\text{THEME}}$ Farrell] **approached** $[^{\text{NP}}_{\text{GOAL}}$ him] $[^{\text{PP}}_{\text{SOURCE}}$ from behind].

{TRANSPORTATION} $[^{\text{NULL}}_{\text{DRIVER}}$] $[^{\text{NP}}_{\text{CARGO}}$ The ore] was **boated** $[^{\text{PP}}_{\text{GOAL}}$ down the river].

Our focus here is on the FrameNet corpus, but another semantically annotated corpus is under development, called the Proposition Bank [10]. This corpus, based on adding semantics to the Penn English Treebank, is projected to soon be larger than FrameNet, and involves comprehensive rather than selective annotation of a corpus. However, it does not incorporate the rich frame typology of FrameNet, and only a somewhat limited role typology; while roles are specified for each verb, there is no generalization across verbs. Finally, Proposition Bank labels only verbs, leaving nouns and adjectives for a later stage; FrameNet includes all three. Since we desire rich semantic information in preference to a large corpus, we use FrameNet annotations as our source of training data. Our methods, however, would generalize to Proposition Bank.

[3] Also, confusingly known as version 0.75 and version 1.0, respectively.

[4] What we call the *predicator* is called the *target* in the FrameNet theory, and what we are calling a *(semantic) role* is called in FrameNet a *frame element*, while what we call a *constituent* or *argument head*, [2] call simply the *head*. We have found that most people find the FrameNet terminology rather confusing, and so have adopted alternative terms here.

2.2 Gildea & Jurafsky's Discriminative model

Gildea and Jurafsky (2002) (henceforth, G&J) were the first to apply a statistical learning technique to the FrameNet data. They describe a discriminative model for determining the most probable role for a constituent given the frame, the predicator and some other features whose description we defer until later in the paper. They evaluate their model on a pre-release version of the FrameNet I corpus, which at that time contained about 50,000 sentences and 67 frame types. Their model was trained by first using the parser of Collins [11], and deriving features from that parse, the original sentence, and the correct FrameNet annotation of that sentence. Their work differs from ours in a number of important respects. Firstly, in all their experiments, they assume that the *frame* is already known, as well as the predicator of interest. While one could certainly imagine first determining the frame from the sentence (for example, one could use the model presented here to do that), their use of a discriminative approach makes it less straightforward to do joint inference over the choice of frame and semantic roles for constituents, as one would wish to do, whereas that is a natural thing to do within a generative model. Secondly, since their discriminative model assigns roles to constituents in the sentence, there is no natural way to handle unexpressed arguments, and they do not attempt to. But unexpressed arguments are common in natural languages, and again are naturally handled in a generative model. Moreover, most of their work assigns roles to constituents individually and independently. Later in their paper, they do develop and consider joint inference over all the semantic roles of a predicator, but this is more naturally done using the kind of model we present here. Finally, although this remains a promissory note, we believe that a generative model will be a better basis for extension via bootstrapping to unlabeled data.

2.3 The Role Labeling Task

With respect to the FrameNet corpus, several factors conspire to make the task of role-labeling challenging, with respect to the features available for making the classification. These results are likely to hold across other theories and methodologies for semantic role determination. The challenges also imply that constructing a hand-built semantic role identifier would prove a daunting task. First, it is not always predictable from the syntactic relationship between two phrases whether they stand in a semantic relationship. Second, many words that may participate in a role have a wide variety of possible roles in which they may participate. There are also many generic roles such as TIME and PLACE that can be indicated by almost any word. Third, the internal structure of a syntactic constituent is not always a good predictor of the role it receives. The prepositional phrase *in the hole*, for example, can be a LOCATION, as in *she sat in the hole*, or a GOAL of movement, as in *she jumped in the hole*. Finally, as mentioned earlier, in many cases roles are null instantiated, which is widespread in many languages; an English example is passive sentences with no specified agent, such as *the cake was eaten*. Thus, the only evidence for the presence of such roles is contextual.

With respect to the relationships between predicators, frames, and roles, further difficulties arise. A leading idea of FrameNet is that there is considerable variety to the semantic role types available in a particular event (for example, PERCEPTION events and COMMERCE events have very different participants). Thus, identifying the frame that is relevant for a particular sentence and predicator narrows the search for roles. However, many predicators are ambiguous with respect to their frame. Further, not all lexical units of a particular frame necessarily have the same distribution of roles. For example, *drop* and *plummet* have lexical entries in the MOTION_DIRECTIONAL frame, but SOURCE is rare for *plummet*, yet quite common for *drop*. As a result, for the task of automatic role assignment a mixture of predicator-specific and frame-specific statistics are potentially useful to deal with sparseness of a particular predicator or role.

3 A Generative Model for Sentence-Role Labeling

Our goal is to identify frames and roles, given a natural language sentence and predicator. As discussed above, G&J's approach to this problem was to determine the most probable role for each constituent of the sentence, given the frame, the predicator and some other features. However, this does not capture null instantiation, or roles that are not reified in the sentence. In addition, a model should ideally capture the relationships between frames and roles, determining which constituents are likely roles for which predicator. To address these concerns we turn to a generative model to determine the sequence of role labels for a sentence. In other words, our model defines a joint probability distribution over predicators, frames, roles, and constituents. While the model is fully general in its ability to determine these variables, in this paper it is only tested on its ability to determine roles and frames when given *both* a list of constituents and a single predicator. The generative model, illustrated in Figure 1, functions as follows. First, a predicator, S, is chosen, which then generates a frame, F. The frame generates

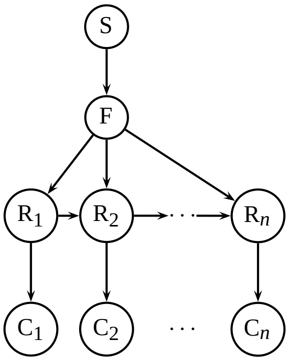

Fig. 1. Role Tagger

a (linearized) role sequence, R_1 through R_n which in turn generates each constituent of the sentence, C_1 through C_n. Note that, conditioned on a particular frame, the model is just a Hidden Markov Model. The sentence-to-constituent mapping is discussed in more detail in Section 3.1.

The model is complicated slightly by the fact that some sentence constituents do not correspond to a labeled semantic role. We handle these constituents with an idea from machine translation: that of the null source. A second complication is the null instantiations, which are also captured by a null, but in this case it is the emission which is a null. Henceforth, null sources will be described by an UNK (unknown) role to avoid confusion with null emissions. We will discuss an example with an unknown role in Section 3.1, and gave an example of a null emission in Section 2.1.

The joint probability for a FrameNet example in this model is

$$P(\mathbf{C}, \mathbf{R}, F, S) = P(S) \times P(F|S) \times P(\mathbf{R}|F, S) \times P(\mathbf{C}|\mathbf{R}, F, S),$$

where $\mathbf{C}$ is the vector of constituent heads, $\mathbf{R}$ is the role vector that generates them, F is the frame, and S is the predicator word. The third and fourth terms of this equation involve sequences. For the role sequence, we usually make a Markov assumption that each word's role is dependent only on the previous role in the sequence. Thus:

$$P(\mathbf{R}|F, S) = \prod_i P(R_i|R_1 \cdots R_{i-1}, F) \approx \prod_i P(R_i|R_{i-1}, F)$$

where the R_i are the roles in the sequence. The Markov assumption has been effective in language modeling and tagging and so seems a good assumption to begin with.

Finally, our basic model assumes that constituent emissions are independent of the frame and predicator given the sequence of roles, that each emission depends only on the role that generated it, and that constituents are independent of each other. Thus:

$$P(\mathbf{C}|F, S, \mathbf{R}) \approx P(\mathbf{C}|\mathbf{R}) = \prod_i P(C_i|R_i),$$

where C_i are the elements of $\mathbf{C}$ and R_i are the corresponding elements of $\mathbf{R}$. This can be compared to a part of speech tagging model where words are independent of each other given the tags, and depend only on the tag in the same position in the sequence. The independence of the constituents and the frame and predicator given the roles seems quite reasonable, given that most roles are frame-specific, and the whole rationale of FrameNet is that frames are sufficiently fine-grained that roles for predicators inside a single frame behave similarly. Adding further dependencies might be expected to only exacerbate the problem of sparseness in the data.

3.1 Training the model

The FrameNet corpus contains annotations for all of the model components described above. To simplify the model, we chose to represent each constituent by its phrasal category together with the head word of that constituent. Since the FrameNet annotations do not include head word information, we determined the heads using simple heuristics. This representation and the method of head-finding are familiar from the statistical parsing literature ([12]). This data then provides a set of constituents with correctly annotated roles for a given sentence, where it is known which constituents correspond to roles and what the appropriate predicator is for those roles. For example, for the example below, the training example would be: S=rode; F=TRANSPORTATION; R_1=DRIVER; C_1=Anne/NP; R_2=VEHICLE; C_2=donkey/NP; R_3=AREA; C_3=on/PP.

{Transportation} “On 26th May $[^{\text{NP}}_{\text{Driver}}$ Anne] **rode** $[^{\text{NP}}_{\text{Vehicle}}$ a donkey] $[^{\text{PP}}_{\text{Area}}$ on the beach],” the letter said .

Most of the parameters for the model are estimated using a straightforward maximum likelihood estimate based on fully labeled training data. Emission probabilities need to be smoothed, due to the sparseness of head words. During training, all words seen only once are replaced by the phrase type label of the constituent of which they are a head. This gives a phrasal-class based model, which is itself smoothed with a uniform phrasal class prior, and the probability of generating unseen words belong to a certain class is estimated as simply a constant (representing $P(word|class)$) times the probability of the phrasal class. Therefore, statistics are gathered both for the probabilities of roles generating each phrase type plus head combination, and there is a backoff model of roles generating a phrase type, and some unknown word within that type.

In an actual semantic parsing application, it would not be known which constituents bear a role of which predicators. We could make use of a syntactic parse in determining constituents that are candidates for roles. In a first approximation of this, we used a parser to determine constituents and their phrase types, and combined these with the FrameNet annotations. For this purpose, we restricted ourselves to training and testing on examples whose annotated predicator is a verb, since these are dealt with in a straightforward manner. The “sentence” level of the model in this case includes only the verb phrase whose head is the predicator, and its subject and arguments. If a constituent is identified in the parse but not in the FrameNet annotation, we label it as an Unk role. Again, this treatment is similar to the case of null emissions in a statistical machine translation model. For this format, the example above would have an additional role inserted at the beginning, with role=Unk and constituent=On/PP.

3.2 Producing the semantic role labels

At inference time, the goal is to produce a sequence of role labels, given a sequence of constituents and a predicator. As just discussed, these constituents may be the head/phrase-type pairs from the FrameNet data, or the head/phrase-type pairs that are the result of parsing a sentence in the corpus and extracting the verb phrase with its subject and arguments. The role-labeling procedure is dependent on the frame, itself a hidden variable at labeling time. If the frame were known, we could simply use the HMM Viterbi algorithm, with the roles as the hidden states and the constituent heads and their phrase type as the emissions. In that case, we would use transition probabilities from only the frame of interest. Because we currently add empty constituents for the null instantiated roles whether using parsing information or not, our Viterbi sequence is of the same length as the input constituent sequence.

For the emission probabilities, there are two options, since a particular role can appear in multiple frames. One option is to condition the emission probabilities also on the frame. That is, the emission probabilities are calculated from only those role/constituent pairs that originally appeared in the given frame. A second option is to calculate emission probabilities for a role over all frames in the training data, since this arguably would provide more evidence and mitigate sparse data problems to some extent. However, the second option also leads to a potential problem, that of words unseen in the given frame but seen as emissions of the role in other frames. We compare both options in the results.

If the frame is not known, the more realistic case, then we have several options. We could just change the model and make the roles a combination of a role and a frame, but then the Viterbi sequence might change frames part way through, which seems unsatisfactory, given the intended semantics of the model. We could marginalize out the frame variable. In practice, given that most roles are particular to individual frames, doing such a marginalization would probably give results little different to our current results, but this also seems conceptually wrong, since we're wanting to do inference for the most likely frame and roles underlying a sentence. So instead we calculate the most probable configuration of all the hidden variables. This generalized Viterbi algorithm is a straightforward instance of max-propagation algorithms for Bayesian networks [13].

For this case, this is equivalent to the less efficient operation of simply finding all frames with $P(F|S) > 0$, compute the role sequence probabilities given the transition probabilities for that frame and the emission probabilities across all frames, and then choosing the maximum product of the prior probability of the frame for the predicator and the probability returned by the HMM Viterbi algorithm.

4 Experimental Results

To test the above model, we trained it on annotated FrameNet data, randomly dividing the data into a training set and an unseen test set. Each frame was randomly split so that 70% of its examples were in the training set and 10% were in the test set. We report on three types of accuracy. First, role labeling accuracy is the number of constituents correctly labeled. Since we label all constituents, this makes the familiar metrics of *recall* and *precision* equivalent. We micro-average by adding up the number of correct labels for *all* examples and dividing by the number of total labels for all examples, so this is not an average accuracy per-sentence, though we have done the calculations both ways, and for these experiments the two figures are quite close to each other. Second, we report the percent of sentences for which all roles are correctly labeled, or full sentence accuracy. Finally, frame accuracy is calculated as the proportion of sentences for which the correct frame was chosen based on the predicator.

For a baseline comparison, we computed the accuracy of a zeroth-order Markov model, treating all transition probabilities between roles as uniform. We also computed the accuracy of choosing, for all constituents, the most common role given the predicator, and the accuracy of choosing the most common role given the frame, where the most common frame ($\arg\max_F P(F|S)$) for the known predicator is chosen.

4.1 Results: Annotated Roles

Our first set of experiments trained and tested our model from the correctly annotated sentences of the FrameNet corpus, together with constituent heads as determined by a parser. We performed most of our experiments on FrameNet I, but ran some experiments with FrameNet II as well.[5] The constituents' heads were chosen by some simple

[5] We regard the FrameNet I results as broadly comparable with those of G&J, though the data sets are not exactly the same, and there are various other differences (we guess the frame

System	Trn Role	Tst Role	Trn Full	Tst Full	Tst Frame
FirstOrder	86.1%	79.3%	75.4%	65.3%	97.5%
ZeroOrder	–	60.0%	–	34.6%	96.5%
BasePredicator	39.9%	39.2%	10.5%	10.2%	N/A
BaseFrame	37.8%	37.6%	9.2%	9.5%	N/A

Table 1. FrameNet I Experimental Results. Key: Role=Role labeling accuracy, Full=full sentence accuracy, Frame=Frame choice accuracy. Trn=Training Set, Tst=Test Set.

heuristics, but their labels correspond to the Phrase Type labels from FrameNet. These tests are similar but not identical to the analysis in Section 4.2 of G&J.

The first results are on 36,805 training sentences, containing a total of 82,169 constituents, and 5299 test sentences containing 11,833 constituents. There are 78 frames, 139 possible role labels, and 1,385 predicators. We obtain 86.1% role labeling accuracy on the training data, 79.3% on the test data. For full sentence accuracy,we obtained 75.4% accuracy on the training data and 65.3% on the test data. Finally, the correct frame was chosen for 98.1% of training sentences and 97.5% of the test sentences. Table 1 summarizes these and the remainder of our results for this data set. We did not measure the training accuracy in the zeroth-order case. These results are roughly comparable to results of 78.5% on test data for G&J's model on data with constituents marked, and they cite a similar result for BasePredicator of 40.6%. We can at least conclude that performance is similar.

We also measured the benefit of exploring all sequences versus only the sequence for the frame with the highest probability given the predicator. The difference is shown in Table 2, for training accuracy only in the First Order and Zero Order case. The differences are about two percentage points in most cases.

System	Role	Full	Frame
First All	79.3%	65.3%	97.5%
First ArgMax	77.2%	63.2%	94.8%
Zero All	60.0%	34.6%	96.5%
Zero ArgMax	58.8%	33.4%	94.8%

Table 2. FrameNet I Arg Max versus all Sequences

Our next set of results are on FrameNet II, where we evaluated only the ArgMax case. Training on 70% and testing on 10% resulted in a corpus of 89,900 training sentences and 12,990 test sentences. Here there are 282 frames, 423 possible role labels, and 4,712 predicators. The performance results on the test set, shown in Table 3, are

whereas they assume it; except in parsing experiments, we use the phrasal category given in FrameNet, whereas they always use phrasal categories returned by a parser, even when using the constituent extent information given by FrameNet). We have recently obtained G&J's data, and hope to provide a more precise comparison in future work.

somewhat weaker than for FrameNet I, but not overly so, considering the increased number of roles and frames.

System	Role	Full	Frame
FirstOrder	73.9%	63.7%	88.7%
ZeroOrder	61.3%	43.0%	89.3%

Table 3. FrameNet II Experimental Results.

In analysis of the role labeling results, we noticed two major sources of error. The first is words unseen in a particular frame but not "rare" over the whole corpus. We could partially address this with a held-out mass for unseen words that is weighted by the prevalence of rare words of each phrase type. Second, some cases are just very difficult, for example, prepositions commonly heading more than one type of role can induce ambiguity, one example being Instrument/Manner ambiguity on *with*-marked roles. We also have difficulties with roles in frames such as Differentiation, which contains roles for Phenomena, Phenomenon1, and Phenomenon2, or Conversation, with its Interlocutors, Interlocutor1, and Interlocutor2 roles. These roles are semantically similar, and we would need a richer syntactic representation to differentiate them.

4.2 Results: All Constituents

In the next set of experiments, we evaluated the system, together with a parser, on the ability to both determine which constituents correspond to roles, and to label those constituents. To do so, we used our statistical parser [14] to parse only the sentences used in the previous section which have a verbal predicator. The parser was trained on Brown and about half of the Wall Street Journal. Our generative model was trained as described above, with the inclusion of UNK roles for constituents not corresponding to a labeled role. At role labeling time, the verb phrases as determined by the parser are presented to the model with (the labeled heads of) their subject and arguments, with the main verb as the predicator. The model now has the option of choosing UNK role labels.

Because of the difficulty in matching parse constituents with their appropriate role labels in the annotated data, the size of the data set for these tests is considerably smaller than that above. We used only the verb phrases corresponding to known frames, but with the UNK roles included. There are are 13,782 training examples, 1,558 test examples, 55 frames, and 980 different predicators. Also, there are 117 unique roles and 43,937 constituents. On this task, the system obtained 81% role labeling accuracy on the training set and 70.1% on the test set. Full sentences were considerably more difficult to get right, with 58.1% training accuracy and 39.5% test accuracy. Frame choice accuracy was 94.5% on the training data and 93.3% on the test data. These results are summarized in Table 4. The only figure G&J give for full sentence accuracy is 38% for a system that had to determine both which constituents correspond to roles, and what those role labels should be, which is again roughly comparable to our 39.5% performance on the test set.

System	Trn Role	Tst Role	Trn Full	Tst Full
FirstOrder	81.0%	70.1%	58.1%	39.5%
ZeroOrder	78.8%	67.8%	50.7%	34%
BasePredicatorParse	35.4%	33.2%	1.0%	0.7%

Table 4. Parse Model Experimental Results.

4.3 Discussion

Our model and these results can be compared and contrasted with those of G&J. Some of the features used by G&J are similar to those used by our model. Both models use the phrase type and head word of each constituent. Both models incorporate the predicator, but in different ways. Our model assumes the predicator is either explicitly given or assumes that each main verb in the sentence is a predicator. A future version could determine the probability that each head word is a predicator.

In addition to these features, G&J introduce several other features. First, the *Governing Category* determines for noun phrases, whether an S or VP most closely dominates the phrase. This feature may provide similar information to that given by our Markov chain. Second, their *Path* feature follows the parse tree from the predicator to the constituent, represented as the string of nonterminals encountered. The final two features missing from our model but present in theirs are whether the main verb phrase of the sentence is in active or passive *Voice*, and the *Position* of the constituent, before or after the predicator. However, these are partially captured by linear order and phrasal constituent type. On the other hand, they always assume knowledge of the frame, and because they only labeled the roles of actual sentence constituents, their model does not include null instantiated roles, nor is it obvious how to extend it to do so.

Finally, our ultimate use for this model is not just role labeling, but to estimate parameters when the training data is only partially observed. In that case, using the maximum likelihood estimate is statistically sound, whereas maximizing the conditional likelihood would not be and a generative model is to be preferred.

5 Conclusion and Future Work

We have described and evaluated a successful generative model for semantic role labeling. Our results to date are encouraging but more remains to be done. While small improvements, such as better unknown word handling, can be made to the model, we also see several larger issues that need to be addressed. To do role boundary detection a more sophisticated model is necessary, since under some circumstances non-verbal predicators assign roles to syntactically non-local constituents. Also, while it is fairly straightforward to generalize the current model to the case of multiple predicators per sentence, an articulated theory of when constituents can take roles from multiple predicators is still under development in FrameNet, and would require further articulation in our theory. Finally, it would also be useful to incorporate some extra syntactic information, such as predicator position, and the presence of coordination, and to model role-shuffling operations such as passivization, imperative forms, and extraposition, since these operations, if not modeled, can obscure linguistically motivated generalizations about the linear order of roles.

Acknowledgments

This work was supported in part by the Advanced Research and Development Activity (ARDA)'s Advanced Question Answering for Intelligence (AQUAINT) Program, and in part by an IBM Faculty Partnership Award to the final author.

References

1. Baker, C.F., Fillmore, C.J., Lowe, J.B.: The Berkeley FrameNet project. In: COLING-ACL. (1998)
2. Gildea, D., Jurafsky, D.: Automatic labeling of semantic roles. Computational Linguistics **28** (2002) 245–288
3. Riloff, E., Jones, R.: Learning dictionaries for information extraction by multi-level bootstrapping. In: Sixteenth National Conference on Artificial Intelligence. (1999) 474–479
4. Nigam, K., McCallum, A.K., Thrun, S., Mitchell, T.M.: Learning to classify text from labeled and unlabeled documents. In: Proceedings of AAAI-98, 15th Conference of the American Association for Artificial Intelligence, Madison, AAAI Press (1998) 792–799
5. Freitag, D., McCallum, A.: Information extraction with HMM structures learned by stochastic optimization. In: Proceedings of AAAI. (2000) 584–589
6. Leek, T.: Information extraction using hidden Markov models. Master's thesis, U C San Diego (1997)
7. Huffman, S.: Learning information extraction patterns from examples. In Wermter, S., Scheler, G., Riloff, E., eds.: Connectionist, Statistical and Symbolic Approaches to Learning for Natural Language Processing. Springer-Verlag (1996) 246–260
8. Resnik, P.S.: Selection and Information: a class-based approach to lexical relationships. PhD thesis, University of Pennsylvania (1993)
9. Rooth, M., Riezler, S., Prescher, D., Carroll, G., Beil, F.: Inducing a semantically annotated lexicon via em-based clustering. In: 37th Annual Meeting of the ACL. (1999)
10. Kingsbury, P., Palmer, M., Marcus, M.: Adding semantic annotation to the penn treebank. In: Proceedings of the Human Language Technology Conference, San Diego, California (2002)
11. Collins, M.J.: Three generative, lexicalised models for statistical parsing. In: ACL 35/EACL 8. (1997) 16–23
12. Collins, M.: Head-Driven Statistical Models for Natural Language Parsing. PhD thesis, University of Pennsylvania (1999)
13. Cowell, R.G., Dawid, A.P., Lauritzen, S.L., Spiegelhalter, D.J.: Probabilistic Networks and Expert Systems. Springer-Verlag, New York (1999)
14. Klein, D., Manning, C.D.: Fast exact inference with a factored model for natural language parsing. In: Advances in Neural Information Processing Systems. Volume 15., MIT Press (2003)