Probabilistic Classifiers for Tracking Point of View

Janyce Wiebe and **Rebecca Bruce**
Computing Research Laboratory
and
Department of Computer Science
New Mexico State University
Las Cruces, NM 88003
wiebe@cs.nmsu.edu, rbruce@crl.nmsu.edu

Appears in the working notes of **Empirical Methods in Discourse**, AAAI Spring Symposia series, Stanford University, March 1995.

Abstract

This paper describes work in developing probabilistic classifiers for a discourse segmentation problem that involves segmentation, reference resolution, and belief. Specifically, the problem is to segment a text into blocks such that all *subjective sentences* in a block are from the point of view of the same agent, and to identify noun phrases that refer to that agent. In our method for developing classifiers (Bruce & Wiebe 1994ab), rather than making assumptions about which variables to use and how they are related, statistical techniques are used to explore these questions empirically. Further, the types of models used in this work can express complex relationships among diverse sets of variables. This work is part of a large project that is in an early stage of development. The contributions of this paper are an illustration of framing a high-level discourse problem in such a way that it is amenable to statistical processing while still retaining its core, and a description of a method for developing probabilistic classifiers that is well-suited for addressing problems in discourse processing.

Introduction

This paper describes work in developing probabilistic classifiers for a discourse segmentation problem that involves segmentation, reference resolution, and belief. Specifically, the problem is to segment a text into blocks such that all *subjective sentences* in a block are from the point of view of the same agent, and to identify noun phrases that refer to that agent.

A *probabilistic classifier* is a system that performs disambiguation by assigning, out of a set of classes, the one that is most probable according to a probabilistic model. The model expresses the relationships among the classification variable and variables that correspond to properties of the ambiguous object and the context in which it occurs. In our method for developing classifiers (Bruce & Wiebe 1994ab), rather than making assumptions about which variables to use and how they are related, statistical techniques are used to explore these questions empirically. Further, the types of models used in this work can express complex relationships among diverse sets of variables. These advantages are particularly important for high-level discourse tasks, for which a great many features and interactions among them seem potentially relevant. This work is part of a large project that is in an early stage of development. The contributions of this paper are an illustration of framing a high-level discourse problem in such a way that it is amenable to statistical processing while still retaining its core, and a description of a method for developing probabilistic classifiers that is well-suited for addressing problems in discourse processing.

POV Tracking

The general problem we address is tracking what Uspensky (1973) calls the *psychological point of view* in narrative. Let *subjective* sentences (Banfield 1982) be ones that present the *private states* of agents (emotions, perceptions, propositional attitudes). The problem of tracking the (psychological) POV is determining, for each sentence, whether or not the sentence is subjective and, if it is, identifying the agent(s) whose POV is taken (the *subjective agent*). What makes this a non-trivial, context-dependent problem is the frequent occurrence of subjective sentences that mention neither the agent nor the state type (belief, intention, etc.). Consider, for instance, the following passage:

> (a.1) John was aware that Mary was incompetent. (a.2) She couldn't even log into the computer.

Sentence (a.2) is easily taken to present a private state of John's, but it leaves the agent and the type of private state entirely implicit. Similarly, consider the following discourse fragment from a non-fictional book about WWII (Irving 1981, p.7):

> (b.1) The young Americans, on their part, had never seen anything like London. (b.2) The buildings were low and gnarled and barnacled with sooty decorations. (b.3) The policemen wore odd helmets, office workers wore bowlers, and

passersby wore blank expressions—no eye contact. (b.4) What shocked them most, perhaps, these kings of the American road, was to find themselves suddenly impotent in the flow of wrong-way traffic.

All of these sentences present private states of the Americans. It is to the Americans that the police helmets are odd, for example. But in (b.2-3), the agents and state types are again implicit.

It is necessary to determine the current POV in narrative in order to distinguish the beliefs of agents from the facts of the narrative, to correctly attribute beliefs and other attitudes to their sources, to recognize agents' intentions, and to understand the discourse relations among sentences. Wiebe has developed an algorithm for tracking POV that decides, for each sentence whether or not it is subjective and, if it is, identifies the subjective agent (Wiebe 1990, 1994). Many of the features used in this algorithm will be incorporated into our current work.

Subjective sentences are ubiquitous in narrative, and there exist many closely related discourse/pragmatic phenomena. In news articles, is the writer presenting what is to be taken as fact, or is s/he presenting the opinion of an agent mentioned in the article? Or, consider discussing someone else's work, say Smith's, in a research paper or text book. After an initial reference to Smith's work, you may go on to describe his or her theory without explicitly saying in each sentence that you are doing so (with a locution such as "In Smith's theory" or "According to Smith"). Further, although not necessarily concerned with private states per se, the same sort of discourse structures arise with what Fauconnier (1985) calls "space builders". Just as a subjective sentence can begin a discourse segment implicitly presenting an agent's POV, an adverbial such as "in 1969" can begin a discourse segment in which subsequent sentences are understood to refer to events that occurred in 1969, even though the date is not subsequently mentioned. An NLP system must be able to recognize such discourse phenomena in order to recover information implicitly communicated in the discourse.

The problem is related to other discourse problems, in that it involves discourse segmentation (Grosz & Sidner 1986) and a task akin to pronoun resolution (identifying the subjective agent). As discussed below, we plan to address a somewhat less fine-grained problem in the interests of feasibility; however, the new problem retains segmentation and addresses interactions between POV tracking and reference resolution.

The Problem

The problem addressed here is a broader segmentation of the text. Rather than deciding upon the POV of each sentence, the task is to segment the text into blocks, each possibly containing both objective and subjective sentences, such that all subjective sentences in a block have the same subjective agent. We can view this task as identifying critical changes, where the text turns from focusing on one agent to focusing on another.

Ideally, we would also like to identify the subjective agent of each block. However, doing so in every case would require unrestricted, general reference resolution, since the subjective agent has to be recovered from the text. Wiebe's POV algorithm identifies the subjective agent intensionally (descriptively) as being, for example, the referent of the subject of the current or a previous sentence, or the subjective agent of the previous subjective sentence, whoever that might be. To arrive at who the subjective agent is, these intensional descriptions are evaluated against the results of an assumed preprocessing stage, which includes reference resolution. Rather than attempting to perform reference resolution automatically in a preprocessing phase, we have elected to make the ultimate goals of the system a segmentation of the text into blocks and, within each block, a judgment as to whether or not the subject of each sentence refers to the subjective agent (whoever he or she is). Each noun phrase successfully judged to be a reference to the subjective agent provides information about him or her and, because all of these noun phrases are co-referential, each contributes information about the referents of all the others as well. For example, "he" tells us the individual is male and "John" identifies his name.

The goals stated above are challenging ones. Thus, we have broken the problem into five steps. The results of each step are available as input to the subsequent steps, and work can proceed on the more challenging steps only once results have been obtained for the simpler steps. The first two phases do not involve POV-segmentation, but instead perform selected syntactic and semantic disambiguation. The latter three are the ones that address POV-segmentation, making successively finer classifications with respect to the identity of the subjective agent. They are probabilistic classifiers (we shall call these "POV classifiers"). The second of the preprocessing phases is also a probabilistic classifier, targeting a key preprocessing requirement of the existing POV algorithm.

In the remainder of this paper, we first discuss preprocessing using existing software and then specify the four classification problems addressed. We then present our method for developing probabilistic classifiers (Bruce & Wiebe 1994ab), discuss related work, and conclude with examples and discussion.

Preprocessing

In a probabilistic classifier, classification is performed with respect to a set of variables corresponding to properties of the ambiguous object and the context in which it occurs (referred to here as the *non-classification variables*, to distinguish them from the *classification variable*, the variable corresponding to the targeted ambiguity). The values of all non-classification variables involved in the classification must be known. This is a particular problem facing automated approaches to high-level discourse processing tasks, which often build upon the results of prior syntactic and semantic analysis. Fortunately, many of the features used in Wiebe's POV algorithm are syntactic and semantic distinctions that can feasibly be at least approximated automatically; these distinctions amount to much less than a full parse or a full representation of the literal meaning of the sentence.

A preprocessing component is being developed to automatically determine the values of a subset of the features used in Wiebe's POV algorithm, as well as others we hypothesize are also relevant (e.g., POS, number, and case). All of these features will be available as candidate non-classification variables for inclusion in any of the classifiers. The component will consist of off-the-shelf software: a POS tagger, a chunker, a name recognizer, a morphological analyzer, and a rudimentary lexicon.

The Classifiers

Developing a probabilistic classifier requires, obviously, that the problem be cast as a classification problem: choosing the objects to be classified and a finite set of mutually exclusive classes for these objects, to serve as the values of the classification variable. Note that the identity of the subjective agent cannot be directly represented as a classification variable. Before a text is processed, the possible subjective agents cannot be known, so cannot be specified as the possible values of the classification variable. The same issue would arise in casting anaphora resolution as a classification problem.

The problems we address are as follows.

Preprocessing classification problem. The objects are the main clauses of sentences. The classification variable is the type of state of affairs that the main clause of a sentence is about. The values are four broad classes drawn from the existing POV algorithm (Wiebe 1990, 1994): private states (e.g., "believe", "hope"), non-private states (e.g., "be", "own"), private actions (e.g., "sigh"), and non-private actions (e.g., "shoot").

First POV classification problem. The objects are sentences. The classification variable has two values: either that the sentence continues the current block (*continues*) or begins a new one (*new*).

Second POV classification problem. The objects are those sentences that begin a new block (i.e., those identified as *new* by POV classifier one). The classification variable values are of the form $\langle sentence, syntacticFunction \rangle$, meaning that the subjective agent is the referent of the noun phrase filling the syntactic function $syntacticFunction$ in sentence $sentence$, e.g., the subject of the current sentence or the subject of the previous sentence.

Third POV classification problem. The objects are the subject noun phrases of the main clauses of the sentences. The classification-variable values are whether or not the noun phrase refers to the subjective agent of the block that it is in.

Graphical Models

To develop a classifier, a statistical model must be formulated that specifies a set of relevant non-classification variables and describes the relationships among these variables and the classification variable.

In this work, we make use of a more general class of models than used in most NLP applications (*graphical models*, Whittaker 1990). Such models are capable of characterizing a rich set of relationships among a large number of variables, where these variables may be of different types as well as spatially separated.

Graphical models are a subset of the class of hierarchical log-linear models and, as such, are well-suited to characterizing and studying the structure of data, that is, the interactions among variables as evidenced by the frequency with which the values of the variables co-occur (Bishop et al. 1975). A log-linear model expresses the probability distribution of a set of variables as a function of the interdependencies that exist among those variables. Formulating such a model involves specifying that certain variables do not interact while allowing the remaining interactions among variables to be arbitrary and unknown.

The probability distribution of a graphical model corresponds to a Markov field. This is because a graphical model is a probability model for multivariate random observations whose dependency structure has a valid graphical representation, a *dependency graph*. The dependency graph of a model is formed by mapping each variable in the model to a node in the graph, and then drawing undirected edges between the nodes corresponding to variables that are interdependent. A valid dependency graph has the property that all variables that are not directly connected in the graph are conditionally independent given the values of the variables mapping to nodes connecting them. For example, if node A separates node B from node C in a dependency graph, then the variable mapping to node B is conditionally independent of the variable mapping to node C, given the value of the variable mapping to node A. Not only does a dependency graph provide a clear interpretation of the interactions among variables, it also has been shown (Pearl 1988) to form the basis of a distributed computational architecture for probabilistic constraint propagation.

An important subset of graphical models is the class of decomposable models. In addition to having a graphical representation, decomposable models possess other desirable properties, two of which are: (1)

there are closed-form expressions for the parameter estimates, and (2) the joint distribution can be expressed as a product of lower-order marginal distributions, thereby reducing the number of parameters that need to be estimated.

Model Formulation

In work developing classifiers for NLP, the assumption is often made that all of the non-classification variables are either conditionally independent given the classification variable or fully independent. When such untested assumptions do not hold, the performance of the classifier will be negatively affected. Of course, all non-classification variables could be treated as interdependent, but, if there are several of them, such a model could have too many parameters to estimate in practice.

Rather than making assumptions about how the variables are related, we explore this empirically. In particular, the goal of the method presented in Bruce & Wiebe (1994ab) is to formulate a decomposable model that describes the relationships among all variables in terms of only the most important interdependencies, that is, the simplest decomposable model that fits the data well. The *fit* of a model is the degree to which the data is approximated by that model.

The first step in the process of formulating a probabilistic model is to select, out of all available non-classification variables, a subset to include in the model. For the POV application addressed in this work, for each classifier, the results of all previous phases are available as potential non-classification variables. Another kind of variable that can be considered is one concerning classifications of previously occurring ambiguous objects (for example, in POV classifier one, whether the tag of the previous sentence is *new* or *continues*). Including this kind of variable is clearly important for discourse processing.

From the set of all potentially relevant variables, we identify those that are most strongly correlated with the classification variable; these are the only non-classification variables included in the model. A variable is considered to be correlated with the classification variable if the model for independence between them fits the data poorly according to the goodness-of-fit test described below. The worse the fit, the more correlated the variables are judged to be.

Having selected a set of variables that are highly correlated with the classification variable, in order to use them in concert to perform classification, the interdependencies among them must also be identified. In our method, this is accomplished via a process of hypothesis testing: patterns of interdependencies among variables are expressed by decomposable models, and then these models are evaluated as to how well they fit the training data. The likelihood ratio statistic, G^2, is used as the measure of the goodness-of-fit of a model. It is distributed asymptotically as χ^2 with degrees of freedom corresponding to the number of interactions omitted from (unconstrained in) the model. The models that are judged to fit are those whose reference χ^2 values are statistically significant. Accessing the fit of a model in terms of the significance of its G^2 statistic gives preference to models with the fewest number of interdependencies, thereby assuring the selection of a model specifying only the most systematic variable interactions.

Once the form of the model has been established, maximum likelihood estimates are used for the parameters of the model. In fact, in this method, parameter estimation is an integral part of the model selection process.

Approximating the joint distribution of all variables with a model containing only the most important systematic interactions among variables limits the number of parameters to be estimated, supports computational efficiency, and provides an understanding of the data. The biggest limitation of the method is that it requires large amounts of hand-tagged data (e.g., in POV classifier one, sentences hand tagged with *new* or *continues*). The validity of the results obtained using this approach are compromised when it is applied to a small data sample, first because maximum likelihood estimates are used for the parameters, and secondly because asymptotic approximations are used for the distribution of G^2.

Model Formulation: Improvements for Large-Scale Applications

We are extending the method described above to better facilitate developing large-scale applications that involve many features and interactions among them. With these extensions, the approach to developing classifiers is as follows. Relevant non-classification variables are selected in the same way as in the original method. Also, as in the original method, we select a decomposable model to express the interdependencies among variables, but, to meet the demands of large-scale applications, we use different methods both for selecting the form of the model and estimating the parameters of that model in order to minimize the amount of hand-tagged data required. These two extensions to the original method are described in the following two sections. It should be noted that the first but not the second of these extensions has been implemented and tested.

Selecting the Form of the Model

We have recently refined the test for model selection so that it is now possible to use only a small sample of tagged data to characterize the form of the model. While our new method for selecting the form of a model still uses the likelihood ratio statistic (G^2) to measure the fit of a model to the data, it eliminates (1) the need for parameter estimation to select the form of the model, and (2) the need to use large-sample approximations of the distribution of G^2. This is accomplished by generating the exact conditional distribution of G^2 in order to determine the significance of the G^2 value describing the fit of a model. The exact conditional distribution of G^2 for each test is the distribution of G^2 values that would be observed for comparable data samples randomly generated from the model being tested. A comparable data sample is one having the same sufficient statistics as the actual training data, where the sufficient statistics are the marginal distributions of the variables that are specified as interdependent in the model. Using the conditional distribution of G^2 values, that is, the distribution of G^2 values calculated for data having the same sufficient statistics as the training data, eliminates the need to estimate the parameters of a model when evaluating its fit. Intuitively, when conditioning on the sufficient statistics, one assumes the dependencies that are specified in the model, and tests for the presence of higher-order interdependencies.

In order to reduce the computational complexity of using exact conditional tests to evaluate the fit of a model, we introduce two simplifications to the procedure. First, the exact significance of the G^2 value for each model will only be approximated by sampling from the conditional distribution of G^2 values as opposed to generating the complete distribution. The significance assigned to the model G^2 value is then equal to the portion of the sampled G^2 values that are at least as large as that of the model.

The second simplification restricts the form of the model that is evaluated in each test. As described above, it is the distribution of G^2 values conditioned on the sufficient statistics of the model being tested that is required to define the significance of a test; unfortunately, it is difficult to generate samples of G^2 values that are conditioned on the sufficient statistics of arbitrarily-complex models. For this reason, model evaluation is conducted as a series of tests, where each test evaluates a single conditional independence between two variables. The complete model is acceptable if all conditional independence relationships expressed in the model are found to be acceptable.

The method just described for selecting the form of a model requires a small amount of sense-tagged data in order to characterize the interdependencies that exist between the various non-classification variables and the classification variable.

Estimating Parameters from Untagged Data

Once the form of the model has been established as just described, it should be possible to obtain reliable estimates of the model parameters from untagged text. The idea is to treat the missing tags as "missing data" and draw on the wealth of statistical techniques for augmenting incomplete data given some knowledge of the form of the complete data density. The scheme we will describe is a stochastic simulation technique referred to as *The Gibbs sampler* (Geman & Geman 1984). It should be noted that there are many possible variations on this general algorithm (Rubin 1991).

The procedure is a Bayesian approach to data augmentation that is very similar in spirit to the EM algorithm (Dempster et al. 1977). The basic idea is straightforward. The observed data, y, is augmented by z, the missing data (i.e., the missing tags), and the problem is set up so that: (1) if the missing data, z, were known, then the posterior density, $P(\theta|y,z)$, could be easily calculated (where θ is the vector of model parameters), and (2) if the model parameters, θ, were known, then the posterior density, $P(z|\theta,y)$, could be easily calculated. An initial value for z, z_1, is assumed and the process iterates between drawing θ_i from $P(\theta|y,z_i)$ and drawing z_{i+1} from $P(z|y,\theta_i)$. The process continues to iterate until the average value for θ, the estimate of the model parameters, converges.

In the Gibbs sampler, the simulation of $P(z|y,\theta)$ is further simplified. This is done by partitioning the missing data as $z = (z_1, z_2, ..., z_n)$ in order to make the drawing of each part of z (i.e., z_j) easier. Specifically, what is done for z is to sample z_1 from $P(z_1|z_2,...,z_n,\theta,y)$, z_2 from $P(z_2|z_1,z_3,...,z_n,\theta,y)$, and so on up to z_n. The Gibbs sampler uses some starting value for z and then generates a value for θ, followed by new values for z_1 through z_n. Each time a new value for a partition of z, say z_j, is generated, the updated values of all other partitions of z are used in selecting that values. As described above, the process iterates between generating θ and generating z_1 through z_n until the average value of θ converges.

The process of stochastic simulation is potentially slow. Hundreds of cycles may be required to achieve reasonable accuracy, with each cycle requiring time on the order of the sum of the number of variables and the number of interdependencies between variables (Pearl 1988). Fortunately, the algorithm can be easily executed in parallel by concurrent processors. A further reduction in computation time can be realized by using simulated annealing as discussed by Geman and Geman (1984).

Performing Disambiguation

A probabilistic classifier assigns to each ambiguous object the class that has the highest probability of occurring with the particular values of the unambiguous variables. Recall that one possible type of variable indicates the classifications of previously-occurring ambiguous objects. Such variables give rise to ambiguities that are interdependent. We are considering using the Gibbs sampler to resolve such ambiguities, i.e., to find the set of classifications that, when taken as a whole, most completely satisfy the constraints among variables. As mentioned above, the complexity of such techniques is a function of the size of the model. If feasible, we will use these techniques. Otherwise, we will assign tags sequentially; that is, in assigning a class to the current ambiguous object, the classifications of previously occurring objects will be treated as known.

Related Work

There has been recent growth in empirically-oriented work in discourse processing. Several researchers have addressed evaluation, investigating the degree to which human subjects agree with one another on discourse tasks (Passonneau & Litman 1993, Hirschberg & Grosz 1992, Hearst 1993). Others have used frequency information to evaluate algorithms. Passonneau and Litman (1993) tagged a corpus with classes and features and tested algorithms hypothesized from the literature. They consider just one feature at a time, so do not address interactions among multiple features.

Some researchers have derived algorithms and models on the basis of frequency information, but have not used systematic, formally characterized methods for developing their models. Aone and McKee (1993), who address anaphor resolution, train their discourse module on a corpus by trying it out with different argument settings. Hirschberg and Litman (1993), who derive a model for cue-phrase disambiguation on the basis of intonational and orthographic features, formally characterize the interactions among the variables that they consider, but their method for deriving their model was manual and not formally characterized. Hearst (1994), who addresses a broad segmentation problem similar to the one addressed here, uses frequency of lexical terms as the criterion by which segmentation is performed. Her method is specific to using this one type of contextual feature.

The most formal approach to developing models that has been applied to discourse problems makes use of decision trees to define the relationships among various contextual features and discourse ambiguities (Siegel & McKeown 1994, Soderland & Lehnert 1994, Litman 1994). The tree construction process partitions the data according to the values of one contextual feature before considering the values of the next, thereby treating all features in each branch of the tree as interdependent. In addition, the tree construction process requires a large amount of tagged data. The method presented here is more flexible, because it eliminates the necessity to treat features as interdependent. Further, as mentioned earlier, our current method requires only a small sample of tagged data to characterize the form of the model, and we are planning to evaluate the method for obtaining estimates of the model parameters from untagged text described earlier.

Examples and Discussion

The following are examples of variables that will be candidates for inclusion in the various classifiers to be developed in this work (more precisely, these are interpretations of such variables). Most are based on features included in Wiebe's POV algorithm; we expect to find, based on experience with that algorithm, that they are indicative of POV. Because the system is to be fully automatic, the values of these variables may not be determined with 100% accuracy. We anticipate that, because our method is probabilistic, such inaccuracies will only be a source of noise that serves to reduce the probabilities assigned to important correlations without obscuring the correct classification.

1. **Variables corresponding to properties of sentences.**
   (a) Whether or not any of a prespecified list of subjective lexical items appears in the sentence (e.g., "amazingly" and "surely");
   (b) whether or not any such items appear in the main clause;
   (c) the type of state of affairs that the main clause is about;
   (d) whether or not the sentence is at the beginning of a paragraph, chapter, or section break;
   (e) the form of the subject noun phrase (pronoun, indefinite description, definite description, and proper name);
   (f) the gender, number, and case of noun phases, as feasible.
2. **Variables concerning classifications of previously occurring ambiguities.**
   (a) How large the current block is so far (one sentence, or a number of sentences above a threshold value);
   (b) the average block size so far (above or below a threshold value);
   (c) the type of state of affairs that the majority of the sentences in the current block are about;
   (d) Whether or not the previous subject noun phrase refers to the subjective agent of the block.

While we expect that the values of each of these variables will be relevant to assigning POV classifications, we do not suppose that the classification problem can be solved by considering each such variable in isolation. Rather, we expect that the problem will require considering sets of variables taken in combination. In fact, empirical investigation of the interactions among variables is a feature of this research that is an important end in itself. Probabilistic models, formulated as described in this paper, characterize the structure of the data; the form of the model identifies interdependencies among important variables, and the parameter estimates provide information about the relationships among the individual values of these variables. For example, consider POV classifier two, in which the classification variable values are of the form $\langle sentence, syntacticFunction \rangle$. Suppose that variable (1.c) (the type of state of affairs that the main clause is about) and variable (1.b) (whether or not subjective lexical items appear in the main clause) are each found to be strongly correlated with the classification variable. Then, the model selection procedure can be used to determine whether or not the values of these three variables (i.e., variables (1.c), (1.b), and the classification variable) are interdependent. If they are found to be interdependent, then the parameter estimates would tell us whether, for example, $\langle currentSentence, subject \rangle$ is the classification with the highest probability of occurring when the value of (1.c) is *private state* and the value of (1.b) is *no* (i.e., there are no subjective elements in the main clause). In fact, this is an example of a correlation we hypothesize we will find.

With the third POV classifier (in which subject noun phrases are classified as referring to the subjective agent or not), we most directly investigate interactions between POV and reference. In general, the subjective agent seems to enjoy a distinguished level of focus. Evidence are cases such as (b.4) at the beginning of this paper. Even without the phrase "these kings of the American road," it would be easy to interpret the first pronoun "them" as referring to the Americans, even though there are numerous competing discourse entities mentioned more recently (the policemen, the passers-by, etc.) (Grosz & Sidner 1986). However, the subjective agent is often referred to non-pronominally as well. Our approach to developing models promises to identify correlations such as those between form of reference and POV.

References

Aone, C. & McKee, D. 1993. A Language-Independent Anaphora Resolution System for Understanding Multilingual Texts. In *Procs. 31st Annual Meeting of the Assoc. for Computational Linguistics (ACL-93)*, pp. 156-163.

Banfield, A. 1982. *Unspeakable Sentences: Narration and Representation in the Language of Fiction.* Boston, MA: Routledge & Kegan Paul.

Bishop, Y. M., Fienberg, S., & Holland, P. 1975. *Discrete Multivariate Analysis: Theory and Practice.* Cambridge: The MIT Press.

Bruce, R. & Wiebe, J. 1994a. A New Approach To Word Sense Disambiguation. In *Procs. 1994 ARPA Human Language Technology Workshop*, pp. 236-241.

Bruce, R. & Wiebe, J. 1994b. Word-Sense Disambiguation Using Decomposable Models. In *Procs. 32nd Annual Meeting of the Assoc. for Computational Linguistics (ACL–94)*, pp. 139-146.

Dempster, A.; Laird, N.; & Rubin, D. 1977. Maximum Likelihood From Incomplete Data Via the EM Algorithm. *Journal of the Royal Statistical Society B* (39): 1–38.

Fauconnier, G. 1985. *Mental Spaces: Aspects of Meaning Construction in Natural Language.* Cambridge, MA: MIT Press.

Geman, S. & Geman, D. 1984. Stochastic Relaxation, Gibbs Distributions and the Bayesian Restoration of Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6: 721–741.

Grosz, B.J. & Sidner, C.L. 1986. Attention, Intentions, and the Structure of Discourse. *Computational Linguistics* 12(3): 175–204.

Hearst, M. 1994. Multi-Paragraph Segmentation of Expository Text. *Procs. 32nd Annual Meeting of the Assoc. for Computational Linguistics (ACL-94)*, pp. 9-16.

Hearst, M. 1993. TextTiling: A Quantitative Approach to Discourse Segmentation. *Technical Report 93/24*, Sequoia 2000 Technical Report, University of California, Berkely.

Hirschberg, J. & Grosz, B. 1992. Intonational Features of Local and Global Discourse Structure. In *Procs. of the Darpa Workshop on Speech and Natural Language.*

Hirschberg, J. & Litman, D. 1993. Empirical Studies on the Disambiguation of Cue Phrases. *Computational Linguistics* 19(3): 501–530.

Irving, D. 1981. *The War Between the Generals.* New York, NY: Congdon & Weed.

Litman, D. 1994. Classifying Cue Phrases in Text and Speech Using Machine Learning. In *Procs. 12th National Conference on Artificial Intelligence (AAAI-94).*

Passonneau, R. & Litman, D. 1993. Intention-Based Segmentation: Human Reliability and Correlation with Linguistic Cues. In *Procs. 31st Annual Meeting of the Assoc. for Computational Linguistics (ACL-93)*, pp. 148-155.

Pearl, J. 1988. *Probabilistic Reasoning In Intelligent Systems: Networks of Plausible Inference.* San Mateo, Ca.: Morgan Kaufmann.

Rubin, D. 1991. EM and Beyond. *Psychometrika* 56(2): 241–254.

Siegel, E. & McKeown, K. 1994. Emergent Linguistic Rules From Inducing Decision Trees: Disambiguating Discourse Clue Words. In *Procs. 12th National Conference on Artificial Intelligence (AAAI-94).*

Soderland, S. & Lehnert, W. 1994. Corpus-Driven Knowledge Acquisition for Discourse Analysis. In *Procs. 12th National Conference on Artificial Intelligence (AAAI-94).*

Uspensky, B. 1973. *A Poetics of Composition.* Berkeley, CA: University of California Press.

Whittaker, J. 1990. *Graphical Models In Applied Multivariate Statistics.* New York, NY: John Wiley & Sons.

Wiebe, J. 1994. Tracking Point of View in Narrative. *Computational Linguistics* 20(2): 233–287.

Wiebe, J. 1990. Recognizing Subjective Sentences: A Computational Investigation Of Narrative Text.

(Ph.D. dissertation) *Technical Report 90-03* Buffalo: SUNY Buffalo Dept. of Computer Science.