Appears in the Proceedings of the Pacific Asia Conference on Expert Systems,
February 11-12 1999, Los Angeles, CA

Integrating Natural Language Subtasks with Bayesian Belief Networks

Ted Pedersen
Department of Computer Science
California Polytechnic State University
San Luis Obispo, CA 93407
tpederse@csc.calpoly.edu

Abstract

The development of automatic natural language understanding systems remains an elusive goal. Given the highly ambiguous nature of the syntax and semantics of natural language, it is often impossible to develop rule–based approaches to understanding even very limited domains of text. The difficulty in specifying rules and their exceptions has led to the rise of probabilistic approaches where models of natural language are learned from large corpora of text. These models usually serve as simple classifiers for particular subtasks such as word sense disambiguation or discourse segmentation. While successful in these limited roles, it is unclear that multiple classifiers can be combined to create comprehensive natural language understanding systems.

Instead, we believe that recent advances in modeling and reasoning with uncertain information offer an appropriate framework for building such systems. We are developing and evaluating new algorithms that learn *Bayesian belief networks* from large corpora of text. These networks will integrate multiple natural language processing subtasks in a single model and will support inferencing mechanisms that go beyond simple classification. We are also developing and evaluating novel sets of features that will allow us to represent and reason with the inherent relationships that exist among natural language processing subtasks.

Introduction

Natural language processing has undergone a transformation in the last decade due to the availability of annotated corpora such as the Brown Corpus and the Penn TreeBank. These are large bodies of online text that have been manually augmented with syntactic and semantic information and can therefore serve as reliable sources of training data for statistical approaches that learn probabilistic models from large corpora of text.

In general, these approaches cast natural language processing subtasks as classification problems. The learned probabilistic models indicate the most likely value for a variable that represents the membership category or classification of an event, given the values of other feature variables that represent the context in which that event occurs. For example, in part–of–speech tagging the classification variable represents the part–of–speech of a particular word and the context in which it occurs is represented by feature variables whose values are the part–of–speech of the immediately preceding words in the sentence. Classification methodologies have been widely applied in natural language processing; word sense disambiguation, parsing, and document classification are but a few examples.

The assumption underlying these approaches is that natural language understanding is decomposable into subtasks that can be independently resolved and merged back together to form larger components. We refer to this as a *bottom–up* model of language understanding. Since each subtask is usually treated independently, interacting and supporting subtasks are assumed to have been resolved before a subtask will be performed. For example, probabilistic classifiers that perform word sense disambiguation usually assume that syntactic ambiguity has been resolved before semantic disambiguation takes place. Likewise, classifiers that segment discourse into coherent blocks often assume that the semantic ambiguity of words has already been resolved. These assumptions result in a bottom–up sequential model of language processing such that syntactic issues must be resolved before semantic processing, which in turn must be resolved before performing discourse–level tasks. This is illustrated in Figure 1, where syntactic, semantic, and discourse level processing are all treated separately. Each box represents a subtask and each enclosed graph represents the relevant features and their interactions within that subtask. Note that the subtasks only interact in a sequential fashion and proceed from low–level syntactic processing to the higher level semantic and discourse processes.

However, in the real world of online text, it is likely that a combination of syntactic, semantic, and dis-

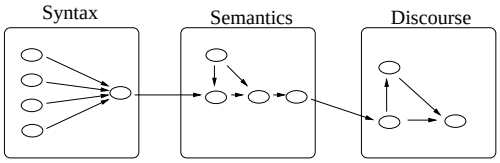

Figure 1: Bottom–up Sequential Language Processing

course level features will be available simultaneously or in an unexpected sequence. For example, if an article is found in the online archives of the baseball section of the sports pages of a newspaper, it seems likely that the general topic of the article is baseball. This information is immediately known and resolves the discourse level subtask of document classification which could then be utilized by other subtasks, given that there is a means to propagate this evidence to them. For example, discourse level information can impact syntactic processing. If baseball is the topic of a document, *bats* can be used as either a noun or a verb (e.g. *Aluminum bats will never be used in major league baseball* versus *He bats fifth in the lineup*). Topic information can also impact semantic processing (e.g., *bats* is unlikely to refer to a mammal if the topic is baseball).

Therefore, we are developing methods of learning Bayesian belief networks that will integrate multiple natural language understanding subtasks into a single unified model. These models will promote the discovery, representation, and utilization of novel interactions among these subtasks. An example of an integrated network is shown in Figure 2, where interactions among features are not restricted to bottom–up relationships but are also top–down, e.g., discourse to syntax, or mixed, e.g., a single semantic feature interacts with both a discourse feature and a syntactic feature.

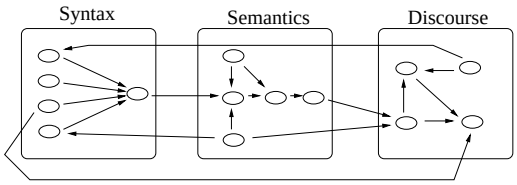

Figure 2: Integrated Network for Language Processing

Bayesian Belief Networks

A probabilistic model of a natural language subtask consists of a set of random variables that represent various lexical, syntactic, semantic, and discourse features, each of which can take on particular values with certain probabilities. The joint distribution assigns probabilities to every possible combination of feature variable values. While the joint distribution supports inferences about any feature in the domain, it becomes intractably large as the number of feature variables increases. In practice it is usually not feasible to specify or obtain evidence for all of the probabilities needed to define a joint distribution.

Bayesian belief networks (Pearl 1988), hereafter simply belief networks, offer a solution to the problems caused by large joint probability distributions. They provide a concise description of joint distributions based strictly on local causal relationships among variables. A belief network is conveniently represented by a graph where the following conditions hold:

1. A set of random variables make up the nodes in the graph.
2. A set of directed edges connects the nodes. The directed edges represent causal influences between variables.
3. Each node has a conditional probability distribution associated with it that quantifies the effect of all the causal influences on a node. The causes are those nodes that have directed edges leading into a node.
4. There are no directed cycles, i.e., it is a directed acyclic graph.

Figure 3 illustrates a simple Bayesian belief network. The structure represents qualitative relationships describing cause and effect relationships among variables; node A is a cause with effect B, node B is a cause with effect D, and node C is a cause with effect B. The conditional distributions associated with this structure are $p(B|A,C)$ and $p(D|B)$. Nodes A and C have unconditional distribution $p(A)$ and $p(C)$ associated with them since they are only causes but never effects. The probability of observing each possible combination of values in these distributions is given by a *parameter* whose value can be specified based upon expert or intuitive knowledge, or learned from training data.

Our objective is to develop new methods for learning the structure and parameter estimates of belief networks for complex and novel integrations of natural language subtasks.

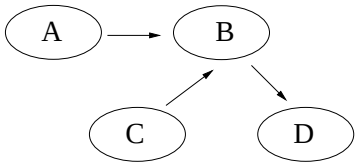

Figure 3: Bayesian belief network

Learning Belief Networks

We will develop and evaluate new methodologies that learn belief networks for natural language processing. There are two scenarios that we are likely to encounter. First, when the training sample of text to be learned from is complete, i.e., when it provides a value for every feature of interest for every observation in the sample, then learning a network structure directly from the training data is feasible. In this case, estimating the values of the parameters in the conditional distributions can be done based strictly on data observed in the training sample using *maximum likelihood estimation*. However, if the sample of text is incomplete, i.e., values are not known or missing for some of the features, then it is usually impossible to learn a network structure directly from the data. Given incomplete training data, it is not possible to directly estimate the parameter values from the data; instead we employ methods that impute values for the incomplete data.

Learning Structure Given Complete Data

The objective in learning a belief network is to discover a network structure that is both a specific representation of the important relationships among the features in the training data and yet still able to generalize to those cases not specifically represented in the training data. This is a challenging problem since the number of possible structures is exponential in the number of features and an exhaustive search of all the possible networks is usually not tractable. We must develop a *search strategy* to guide the learning algorithm through the space of possible networks and an *evaluation criterion* to measure the acceptability of a network, usually in terms of how closely the network characterizes or *fits* the training data.

Our previous work ((Pedersen, Bruce, & Wiebe 1997), (Pedersen & Bruce 1997b)) utilized sequential search strategies and information criteria to select probabilistic classifiers for word sense disambiguation. Here we extend those methodologies to learn belief networks that will support inference on any variable in the domain, not just a single classification variable.

Sequential Search Strategies A sequential search adds (or removes) interactions among features in steadily increasing (or decreasing) levels of complexity, where complexity is measured in terms of the number of interactions in the network. Adding interactions to simple networks is known as *forward inclusion* while removing them from complex networks is *backward elimination*. This research will develop methods of forward inclusion for learning belief network structures. Backward elimination is problematic since the search strategy begins with complex networks that have large conditional probability distributions with many parameter values to estimate. It is difficult to obtain sufficient evidence from the training sample to support estimates for the networks that must be evaluated early in a backward search.

However, determining the initial structure with which to begin forward inclusion poses a dilemma. Forward searches often begin with the model of independence, a structure where there are no interactions among features. Edges are added one at a time until a structure is found that balances complexity and fit. However, early in the search the impact of adding interactions to the network are evaluated relative to a very small number of other interactions which can result in bypassing more complex interactions that may not be apparent in the limited contexts available early in forward search. We will take three approaches to address the limitations of forward search:

1. Initialize forward searches with a limited amount of expert knowledge. Begin the search with a structure that includes interactions that are well supported by other studies or are commonly acknowledged.

2. Initialize forward searches with fixed structures that are variants of the Naive Bayesian classifier. This is a network structure where there are no direct interactions among the feature variables and where all feature variables interact with a single classification variable, i.e., the feature variables are all causal nodes and the classification variable is the only effect node in the network.

3. Randomly initialize the network structure with a fixed number of interactions and then perform a combination of backward and forward search to arrive at a selected network. This process will be repeated a number of times and the results from each stage will be combined into a composite network. This is a variation on the idea of *model averaging* (e.g., (Madigan & Raftery 1994)), where averaged or composite networks are learned as opposed to selecting a single best–fitting network.

Information Criteria The degradation and improvement in fit of candidate networks relative to the current network is assessed by an evaluation criterion. We propose to investigate the applicability of a general class of methods known as the *information criteria* to belief network learning. We will focus on Akaike's Information Criteria (AIC) (Akaike 1974) and the Bayesian Information Criteria (BIC) (Schwarz 1978). These criteria are based on the log–likelihood ratio G^2, a frequently used test statistic that measures the deviance between what is observed in the data and

what would be expected to be observed, if the network under evaluation adequately characterizes or fits the data.

Learning Parameter Estimates Given Incomplete Data

If the training data is incomplete, then learning a network structure is often not possible. In such cases we will rely upon expert or intuitive specification of the structure. However, the task of estimating the parameter values of the conditional probability distributions remains. Generally, if the network structure is given and there are missing values in the training data, we can employ methods that will impute values for the missing data and then make estimates for the parameters based on those imputed values.

Two popular methods of imputing values for missing data are the Expectation Maximization (EM) algorithm (Dempster, Laird, & Rubin 1977) and Gibbs Sampling (Geman & Geman 1984). We have used both in our previous work ((Pedersen & Bruce 1997a)) to learn parameter estimates for Naive Bayesian classifiers that were applied to word sense disambiguation. Here we extend their use to belief networks for integrated natural language processing.

The EM algorithm formalizes a traditional method of handling missing data, starting with a guess of the initial values of the parameters. Thereafter, the following steps are performed iteratively: (1) replace missing data values by their expected values given the guessed parameters, (2) estimate parameters assuming that the missing data is given by the expected values, (3) re–estimate the missing values assuming the new parameter estimates are correct, and (4) re–estimate the parameters assuming the new missing values are correct, iterating until these estimates converge at a maxima.

Gibbs Sampling can be cast as a stochastic version of the EM algorithm. A Gibbs Sampler iterates much as the EM algorithm except that it replaces missing data values and re–estimates parameters via repeated sampling from conditional distributions defined by the network structure whereas the EM algorithm simply maximizes these distributions. Chains of estimates are generated during Gibbs Sampling for each parameter. These chains will eventually converge to a stationary distribution.

While the EM algorithm is known to converge rather quickly to parameter estimates associated with missing data, it is susceptible to getting stuck in local maxima. Gibbs Sampling is guaranteed not to get stuck in local maxima but can be very slow to converge. Therefore, we are developing a hybrid approach that initializes the Gibbs Sampler with the parameter estimates found by the EM algorithm. This overcomes the potential danger of the EM algorithm arriving at a local maxima while helping speed convergence of the Gibbs Sampler.

Integrated Networks of Natural Language

In the previous section we outline a methodology for learning belief networks. Our intention is to develop belief networks that integrate a variety of natural language processing subtasks. In this research we will focus on the integration of three key subtasks: word sense disambiguation, subtopic shift identification, and document classification.

Word Sense Disambiguation

Word sense disambiguation is a central problem in natural language processing; it is the process of selecting the most appropriate meaning for an ambiguous word, given the context in which it occurs. It is not yet well understood what constitutes the necessary and sufficient context to disambiguate a word; in fact, the representation of context is often the salient difference among approaches to this problem.

Corpus–based approaches have generally relied upon a window of surrounding words to provide context. A window of 100 words was suggested in (Gale, Church, & Yarowsky 1992), although small windows of one or two surrounding words have also proven effective (e.g., (Ng & Lee 1996)). Another commonly used representation of context is the so–called *bag of words*, where each word that occurs in the training sample is represented by a feature variable (e.g., (Mooney 1996)). Syntactic structure has also proven useful. For example, the part–of–speech of surrounding words are common representations of context (e.g. (Bruce & Wiebe 1994)) as are verb–object structure (e.g., (Ng & Lee 1996)).

Thus, current approaches to word sense disambiguation generally focus on syntactic and lexical representations of context. However, the *one–sense–per–discourse* hypothesis (Gale, Church, & Yarowsky 1992) holds that content words will largely be confined to one sense when they appear in specific domains. Despite the intuitive appeal of this hypothesis, discourse level features are generally not included in probabilistic word sense disambiguation algorithms. We believe that this is at least partially due to the bottom–up sequential methodology that classification based approaches to language processing seem to impose. We are optimistic that a belief network that allows evidence from the discourse level to impact semantic and syntactic processing will result in improved performance at all levels of language processing.

Subtopic Shift Identification

Establishing the boundaries of shifting subtopics in text is an important area of research in discourse analysis and information retrieval. The object of this subtask is to identify and mark locations in a document where the subtopic changes to a measurable degree. These markings will form a contiguous, non–overlapping series of subtopic shifts. However, this does not include identifying the nature of the subtopic, just indicating that it has shifted and whether or not it has returned to a previously mentioned subtopic or if a new one has been introduced.

A variety of fairly obvious structural features exist in certain kinds of text such as headings and sub–headings that serve as markers for subtopic shifts. However, there are many types of text where this information is not available and the identification of subtopic shifts is a non–trivial problem.

The underlying premise to most approaches to this problem is that changes in the distribution and occurrence of content words in a text will signal changes in subtopic (e.g., (Hearst 1997)). Several well known statistical tests have been employed to identify significant changes in vocabulary, among them the log–likelihood ratio G^2 and the t–test. We also believe that features from semantic level subtasks such as word sense disambiguation can be of benefit for subtopic shift identification.

Document Classification

The ability to divide a collection of documents into pre–defined subject categories is a very practical and important application for natural language processing and information retrieval, particularly given the large amount of text that is now available online. This problem is usually approached by training a learning algorithm with examples of documents that have been manually categorized into classes or broad topics. These methods make discrimination decisions based upon the appearance or distribution of content words in documents, as well as on various metrics that define *semantic distance* between documents.

Perhaps the most common approach is to represent documents using variants of the *bag of words* feature set. Each document is represented by a single vector where each feature in the vector is a binary variable indicating the presence or absence of a certain word from the document or query. This representation of context has been widely used to create Naive Bayesian classifiers. Recent approaches require smaller amounts of training data and do not necessarily include all the content words in the training sample in the model (e.g., (Koller & Sahami 1997)).

A number of other approaches have come from information retrieval research. For example, the vector space model (Salton & McGill 1983) has been applied to document classification. In general, each word in a document is treated as an axis in a highly dimensional space. All documents in a training sample that represent a particular category are plotted in this space, with the distance along each axis dependent on the number of times each word occurs in the training sample. To determine the category membership of a new document, one calculates the cosine of the angle between the clusters of points representing the various categories of documents and the cluster representing the document to be classified. The category of the new document will be that which has the cluster of points that lie closest to the cluster associated with the new document.

Integration of Subtasks

Our objective is to create integrated belief networks that represent all of the features and interactions that exist in and among word sense disambiguation, subtopic shift identification, and document classification. We believe that there are inherent and important interactions among these subtasks that are not discovered, represented, or utilized by current methodologies.

However, in order to learn integrated models, training data must be available for all of the subtasks. We propose to develop large sets of such data by taking advantage of naturally occurring training examples. Online text, especially as found in hyper–linked environments, contains a great deal of contextual information that goes beyond what is immediately contained in the text; documents at Web sites are often organized by topic, labels on hyper–links can provide important information as to semantic content, etc.

We will extract articles from freely available sources such as the online archives of wire services and newspapers where those articles have already been categorized by topic. This will give us a large and ready–made source of training data for the document classification subtask.

Within the document classification training data, we will identify and save only those articles that have headings and sub–headings included as a part of the text. We will treat these not as text but simply as markers of subtopic shifts. We now have a corpus of training data that includes a large number of document classification and subtopic shift examples.

Obtaining training data for word sense disambiguation is more difficult since there are few naturally occurring sources of disambiguated text. While there are manually sense–tagged corpora available, they are

generally not of sufficient breadth to provide adequate quantities of training data for document classification and subtopic shift as well. Therefore, we will create sense–tagged text using the *pseudo–word* methodology ((Gale, Church, & Yarowsky 1992), (Schutze 1993)). Rather than manually annotating different instances of a word with sense indicators, this approach combines two unrelated words and creates a new single word that is ambiguous; the possible meanings are those of the individual words. For example, all instances of *apple* and *baseball* in a corpus are combined into a new word, *apple–baseball*. The pre–combination version of the corpus serves as a gold standard by which the automatic disambiguation of *apple–baseball* can be evaluated. This is an efficient means of creating training data that has the further advantage of providing a reliable gold standard for evaluation; human disambiguation tends to be somewhat unreliable as well as time–consuming.

References

Akaike, H. 1974. A new look at the statistical model identification. *IEEE Transactions on Automatic Control* AC-19(6):716–723.

Bruce, R., and Wiebe, J. 1994. Word-sense disambiguation using decomposable models. In *Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics*, 139–146.

Dempster, A.; Laird, N.; and Rubin, D. 1977. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society B* 39:1–38.

Gale, W.; Church, K.; and Yarowsky, D. 1992. A method for disambiguating word senses in a large corpus. *Computers and the Humanities* 26:415–439.

Geman, S., and Geman, D. 1984. Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 6:721–741.

Hearst, M. 1997. TextTiling: Segmenting text into multi–paragraph subtopic passages. *Computational Linguistics* 23(1):33–64.

Koller, D., and Sahami, M. 1997. Hierarchically classifying documents using very few words. In *Proceedings of the 14th International Conference on Machine Learning*, 170–178.

Madigan, D., and Raftery, A. 1994. Model selection and accounting for model uncertainty in graphical models using Occam's Window. *Journal of American Statistical Association* 89:1535–1546.

Mooney, R. 1996. Comparative experiments on disambiguating word senses: An illustration of the role of bias in machine learning. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 82–91.

Ng, H., and Lee, H. 1996. Integrating multiple knowledge sources to disambiguate word sense: An exemplar-based approach. In *Proceedings of the 34th Annual Meeting of the Society for Computational Linguistics*, 40–47.

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

Pedersen, T., and Bruce, R. 1997a. Distinguishing word senses in untagged text. In *Proceedings of the Second Conference on Empirical Methods in Natural Language Processing*, 197–207.

Pedersen, T., and Bruce, R. 1997b. A new supervised learning algorithm for word sense disambiguation. In *Proceedings of the Fourteenth National Conference on Artificial Intelligence*, 604–609.

Pedersen, T.; Bruce, R.; and Wiebe, J. 1997. Sequential model selection for word sense disambiguation. In *Proceedings of the Fifth Conference on Applied Natural Language Processing*, 388–395.

Salton, G., and McGill, M. 1983. *An Introduction to Modern Information Retrieval*. New York: McGraw–Hill.

Schutze, H. 1993. Word space. In Hanson, S.; Cowan, J.; and Giles, C., eds., *Advances in Neural Information Processing Systems 5*. Morgan Kaufmann Publishers.

Schwarz, G. 1978. Estimating the dimension of a model. *The Annals of Statistics* 6(2):461–464.