Integrating Symbolic and Statistical Approaches in Speech and Natural Language Applications

Marie Meteer & Herbert Gish

BBN Systems & Technologies
Cambridge, Massachusetts
mmeteer@bbn.com
hgish@bbn.com

ABSTRACT

Symbolic and statistical approaches have traditionally been kept separate and applied to very different problems. Symbolic techniques apply best where we have a priori knowledge of the language or the domain and where the application of a theory or study of selected examples can help leverage and extend our knowledge. Statistical approaches apply best where the results of decisions can be represented in a model and where we have sufficient training data to accurately estimate the parameters of the model. Another factor in selecting which approach to use in a particular situation is whether there is sufficient uncertainty to warrant the need to make educated guesses (statistical approach) rather than assertions (symbolic approach).

In our work in gisting, word spotting, and topic classification, we have successfully integrated symbolic and statistical approaches in a range of tasks, including language modeling for speech recognition, information extraction from speech, and topic and event spotting. In this paper we outline the contributions and drawbacks of each approach and illustrate our points with the various components of our systems.

1. INTRODUCTION

Symbolic and statistical approaches have both made significant contributions in speech and natural language processing. However, they have traditionally been kept separate and applied to very different kinds of problems. Most speech recognition systems use statistical techniques exclusively, whereas natural language (NL) systems are mostly symbolic. We are seeing more integration of statistical methods in NL, but usually in some well defined component, such as a statistically based part of speech tagger as a preprocessor to parsing.

In this paper, we have two goals: first, to characterize the kinds of problems that are most amenable to each of these approaches and, second, to show how we have integrated the approaches in our work in information extraction from speech, topic classification, and word and phrase spotting. We begin with a brief overview characterizing the two approaches, then discuss in more detail how we have integrated these two approaches in our work.

1.1 Characterizing symbolic and statistical approaches

Symbolic approaches have dominated work in NL. By writing rules, we can take advantage of what we already know about a language or domain and we can apply a theoretical framework or study of selected examples to leverage and extend our knowledge. Most symbolic approaches also have meaningful intermediate structures that indicate what steps a system goes through in processing. Furthermore, since in a rule based approach the system either works or fails (as opposed to being more or less likely as is the case in a statistical approach), we generally have a clearer understanding of what a system is capable of and where its weaknesses lie. However, this feature is also the greatest flaw of this kind of approach, as it makes a system brittle.

Statistical approaches begin with a model and estimate the parameters of the model based on data. Since decisions are more or less likely (rather than right or wrong), systems using these approaches are more robust in the face of unseen data. In particular, statistical modeling approaches provide the conditional probability of an event, which combines both prior knowledge of the distribution of events and the distribution learned from a training set, which can take into account both how often an event is seen and the context in which it occurs. There are two important considerations in choosing to use a statistical approach: (1) the output must be representable in a model--that is, we need to understand the problem well enough to represent output and specify its relationship to the input. This can presently be done for part of speech tags, for example, but not for discourse; (2) there must be sufficient data (paired I/O) and/or prior statistical knowledge to estimate the parameters.

While these approaches have been kept separate, they have influenced each other. Statistical techniques have brought to NL a clearer notion of evaluation: that there are separate training and testing corpuses and a "fair" test is on data you haven't seen before. Symbolic techniques have brought the notion of understanding a problem by looking closely at the places where it performs poorly. For example, we're seeing a renewed emphasis on tools in speech processing work.

1.2 Integrating symbolic and statistical techniques

In determining how to most effectively combine these approaches, it is useful to view them not as a dichotomy, but rather as a continuum of approaches. Kanal and Chandrasekan (1972) take this view in their analysis of pattern recognition techniques, which they characterize as, at one end, purely "linguistic", with generative grammars representing syntactic structure, and at the other "geometric" approaches, which are purely statistical--patterns are represented as points in a multidimensional feature space, where the "features" are left undefined in the model. In the middle are "structural" approaches, where patterns are defined as relations among a set of primitives which may or may not be associated with probabilities.

Kanal and Chandrasekan argue that rather than select a linguistic or geometric solution for a particular problem, one should divide the problem into subproblems hierarchically, deciding at each level whether to apply a solution from the range between linguistic and geometric or to further subdivide. In this view the various methods are complementary, rather than rivals. Important considerations in making the choice of what approach to use is how much and what kind of a priori information is available and where information is noisy or uncertain.

In fact, nearly all "statistical" approaches used in NL and speech fall somewhere in this continuum, rather than at the extreme. Purely statistical topic classification techniques use words as the primitives, which are features that have some meaning and relationships to one another, even though these relationships may be exploited only through statistical correlations. The states in a hidden Markov model for speech form phonemes, which is conceptual rather than acoustic phenomenon and specific to a particular language, and the expansion of phoneme states into networks are based on a dictionary. Therefore, even in a null grammar there is a great deal of a priori knowledge being brought to bear.

In the work described here, we have attempted a close integration of statistical and symbolic methods that leverages the a priori knowledge that can be represented in phrase grammars with the knowledge that can be acquired using statistical methods. For example, a classification algorithm can select which key words can be used to discriminate a topic. By adding semantic features to a text using a parser and semantic grammar, we can increase the amount of domain specific information available for the classification algorithm to operate over. Another example is in language modeling for recognition: a statistical N-gram language model provides information on the likelihood of one word to follow another; by adding phrase grammars, we can also learn the likelihood of particular domain specific phrases, and then we can use that same grammar to actually interpret those phrases and extract the information being communicated. The body of this paper describes in detail where we have chosen to integrate linguistic and structural knowledge into our statistical algorithms.

2. APPLICATION OF TECHNIQUES

The bulk of our work in integrating symbolic and statistical approaches has been in the development of the "Gister" system (Rohlicek, et. al 1992), which is designed to extract information from voice communications. We developed and tested the algorithms using off-the-air

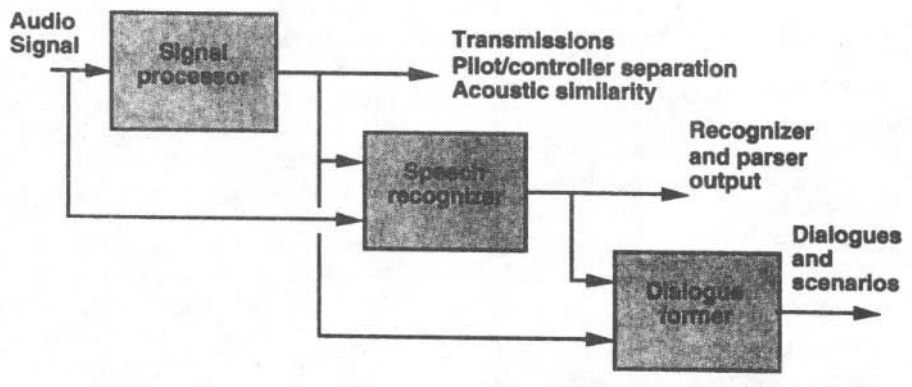

Figure 1: The high level "Gister" boxology

commercial air traffic control recordings, where the goal was to identify the flights present and determine the scenario (e.g. takeoff, landing). We have also extended the system to extract more specific information from the ATC commands, such as direction orders and tower clearances. Figure One shows the overall boxology of the Gisting system.

There are several characteristics of the domain that make it amenable to the unique combination of techniques we employed. First, the language is stereotypical, that is there are few variations in the way the information can be expressed, and we have available expertise on how the information is expressed: it is regulated by the FAA and described in FAA manuals. Second, the signal is very noisy, so that traditional techniques don't work very well (recognition results show a 25-30% word error rate). Our goal was to leverage our a priori knowledge of the domain to reduce the uncertainty inherent in the problem. The most obvious place to start was to improve speech recognition by introducing domain specific information in the language model.

2.1 Language Modeling

The role of the language model in the speech recognition component is to constrain the possibilities of what word can come next and to mark each possibility with its probability: the likelihood that it will occur in a particular context. A common approach to language modeling is to use statistically based Markov-chain language models (n-gram models). While this approach has been shown to be effective for speech recognition, there is, in general, more structure present in natural language than n-gram models can capture. In particular n-grams do not explicitly capture long distance dependencies. For example, a private plane identifier consists of the name of a plane type, some digits, and one or two letter words (e.g. "Sessna six one two one kilo"). Because of the frequency of digits in this domain, an n-gram will find that the most likely thing to follow a digit is another digit; the relationship between the first elements of the phrase (the plane type) and the last (a letter word) is lost.

In our approach we integrated phrase grammars (which were already being used to extract information from the results of recognition) with n-grams, thereby introducing as much linguistic structure and prior statistical information as is available while maintaining a robust full-coverage statistical language model for recognition.

As shown in Figure Two, there are two main inputs to the model construction portion of the system: a transcribed speech training set and a phrase-structure grammar. The phrase-structure grammar is used to partially parse the training text. The output of this is: (1) a top-level version of the original text with subsequences of words replaced by the non-terminals that accept those subsequences; and (2) a set of parse trees for the instances of those nonterminals. We first describe the parser and grammar and then discuss how we use them for language modeling.

For both the language modeling and information extraction (the shaded boxes in Figure 2), we are using the partial parser Sparser (McDonald 1992). Sparser is a bottom-up chart parser which uses a semantic phrase structure grammar (i.e. the nonterminals are semantic categories, such as HEADING or FLIGHT-ID, rather than traditional syntactic categories, such as CLAUSE or NOUN-PHRASE). Sparser makes no assumption that the chart will be complete, i.e. that a top level category will cover all of the input, or even that all terminals will be covered by categories, effectively allowing unknown words to be ignored. Rather it simply builds constituent structure for those phrases that are in its grammar.

Our approach to creating the rules was typical of symbolic approaches: we wrote rules using our knowledge of the ATC domain gained from experts and manuals, ran them on a portion of our data, inspected the results, rewrote the rules, and iterated. In the case of flight IDs, we could apply more extensive evaluation techniques since each utterance in our corpus was already annotated with this

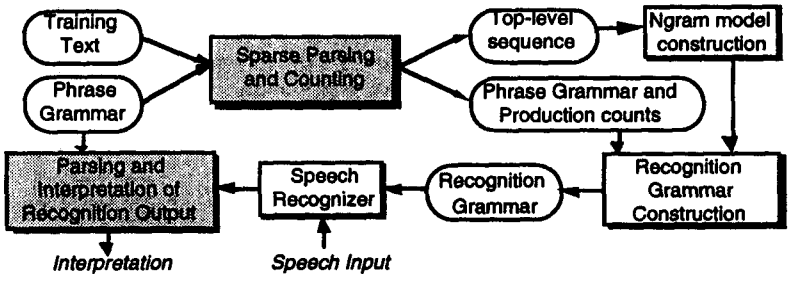

Figure 2: Language Modeling and Information Extraction in the Gisting system

information. However, for other kinds of statements, such as controller orders or pilot replies, there was no master "answer" list against which to evaluate. We had only two measures to use to evaluate our grammar:[1] the overall coverage (what percentage of the words was covered by some category in the grammar), and the specific coverage, which can only be determined by inspecting the results by hand and noticing when some command occurred that was not picked up by the parser. Note that since in this domain we know that there is relatively little variation, so that sampling the data can be assumed to be sufficient to determine coverage, which is not the case in less constrained domains. Figure 3 shows a small set of examples of the rules:

```
R1 (def-rule land-action > ("land"))
R2 (def-rule takeoff-action > ("takeoff"))
R3 (def-rule takeoff-action > ("go"))
R4 (def-rule clrd/land > ("cleared" "to" land-action)
R5 (def-rule clrd/takeoff > ("cleared" "to" takeoff-action))
R6 (def-rule clrd/takeoff > ("cleared" "for" takeoff-action )))
R7 (def-rule tower-clearance > (runway clrd/land)
R8 (def-rule tower-clearance > (runway clrd/takeoff ))
```

Figure 3: Phrase structure rules for tower clearance

The n-gram model was trained not with the original transcripts, but rather with transcripts where the targeted phrases defined in our grammar were replaced by their nonterminal categories. Note that in this case, where goal is to model aircraft identifiers and a small set of air traffic control commands, other phrases like the identification of the controller, traffic information, etc., are left as words to be modeled by the n-gram. Examples of the original transcripts and the n-gram training are shown below:

>Nera twenty one zero nine runway two two right cleared for takeoff

>COMMERCIAL-AIRPLANE TOWER-CLEARANCE

>Nera thirty seven twelve Boston tower runway two two right cleared for takeoff

>COMMERCIAL-AIRPLANE Boston tower TOWER-CLEARANCE

>Jet Link thirty eight sixteen Boston tower runway two two right cleared for takeoff traffic on a five mile final landing two two right

>COMMERCIAL-AIRPLANE Boston tower TOWER-CLEARANCE traffic on a five mile final landing RUNWAY

>Jet Link thirty eight zero five runway two two right cleared for takeoff sorry for the delay

>COMMERCIAL-AIRPLANE TOWER-CLEARANCE sorry for the delay

Figure 4: Training text modified by parser

For the specific phrases we are interested in, we use the parse trees are used to obtain statistics for the estimation of production probabilities for the rules in the grammar. Since we assume that the production probabilities depend on their context, a simple count is insufficient. Smoothed maximum likelihood production probabilities are estimated based on context dependent counts. The context is defined as the sequence of rules and positions on the right-hand sides of the rules leading from the root of the parse tree to the non-terminal at the leaf. The probability of a parse therefore takes into account that the expansion of a category may depend on its parents.

For example, in the above grammar (Figure 3), the expansion of TAKEOFF-ACTION may be different depending on whether it is part of rule 5 or rule 6. Therefore, the "context" of a production is a sequence of rules and positions that have been used up to that point, where the "position" is where in the RHS of the rule the nonterminal is. For example, in the parse shown below (Figure 4), the context of R2 (TAKEOFF-ACTION > "takeoff") is rule 6/position 3, rule 8/position 2. (See Meteer & Rohlicek 1993 for a more detailed discussion of the probabilities required evaluate the probability of a parse.)

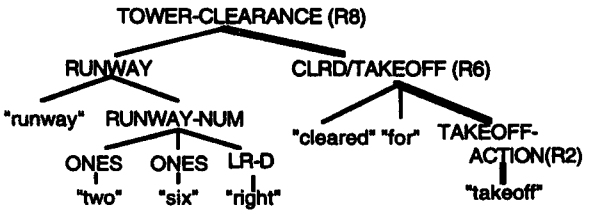

Figure 5: Parse tree with path highlighted

In order to use a phrase-structure grammar directly in a time-synchronous recognition algorithm, it is necessary to construct a finite-state network representation.[2] If there is no recursion in the grammar, then this procedure is straightforward: for each rule, each possible context corresponds to a separate subnetwork. The subnetworks for different rules are nested. Figure 6 shows the expansion of the rules in Figure 3.

[1] Note that given the narrowness of the domain, the issue in processing transcripts is rarely correctness, but rather coverage: do the rules capture all of the alternative ways the information can be expressed.

[2] The phrase grammar formalism is context free; however, in practice, we limited the grammar to finite state so that it can be more easily integrated into the recognizer. We are considering various means of finite state approximations in order to use a more powerful grammar, but haven't found sufficient need in this domain to press the issue.

There have been several attempts to use probability estimates with context free grammars. The most common technique is using the Inside-Outside algorithm (e.g. Pereira & Schabes 1992, Mark, et al. 1992) to infer a grammar over bracketed texts or to obtain Maximum-Likelihood estimates for a highly ambiguous grammar. However, most require a full coverage grammar, whereas we assume that only a selective portion of the text will be covered by the grammar. A second difference is that they use a syntactic grammar, which results in the parse being highly ambiguous (thus requiring the use of the Inside-Outside algorithm). We use a semantic grammar, with which there are rarely multiple interpretations for a single utterance in this domain.

2.2 Information extraction

The information extraction component of the system employs purely symbolic techniques, using same grammar defined for language modeling (as in Figure 3) with associated routines for creating referents as a side affect of firing a rule. Since the uncertainty of the problem lies in the fact that the recognition is errorful, once a grammar has been developed on one set of transcripts one can achieve nearly perfect extraction of flight IDs and commands, since they are the most regular (and regulated) portions of the utterances. In fact, because of this, we were able to use the results of the parser on the transcripts to provide an answer key for evaluation. While this is not a completely accurate test, since there may be cases where a command is expressed in a way that is outside the competence of the grammar, it does make evaluation tractable, since the time it would take to mark the transcripts by hand would be prohibitive. (See Meteer & Rohlicek 1994 for a more detailed description of the information extraction portion of the system and the precision and recall results.)

2.3 Scenario classification

Another component of the Gisting system is scenario classification: given a dialog between pilot and controller, determine the overall scenario being followed. An important aspect of the problem is that classification is performed on the output of the speech recognizer. We used a standard statistical technique for classification, a decision tree constructed using the CART methodology (Breiman, et.al). Decision trees have the advantage that they simultaneously select what the most discriminating features are (from some given feature set, which in the case of text classification is generally the words), and build the model.

Decision trees are interesting predictors, in that they often find features that are telling, but that an expert would not necessarily have thought of. For example if one scenario is more likely to include a radio frequency, then the word "point" may turn out to be very discriminating. When applying classification to the output of recognition, one must choose not only features that are distinguishing, but also ones that are easily recognized, so that they will be reliably in the output. One must be also careful to cross validate results on a test set to avoid overtraining: finding features that are peculiar to the training. For example if in the collected data, one airline had many more takeoffs than landings, then that airline may be picked as a discriminator, even though it is not a good discriminator in general (all the planes that take off eventually land .

We used integrated symbolic methods into classification by using the parser and grammar to augment the input to the classifier with semantic features, as shown in the example below.[3] This is the same process as that which created the

[3] Note that for clarity this example is from the transcriptions, not from the output of recognition. In the Gisting system, the classifier is trained on both the annotated ten best outputs

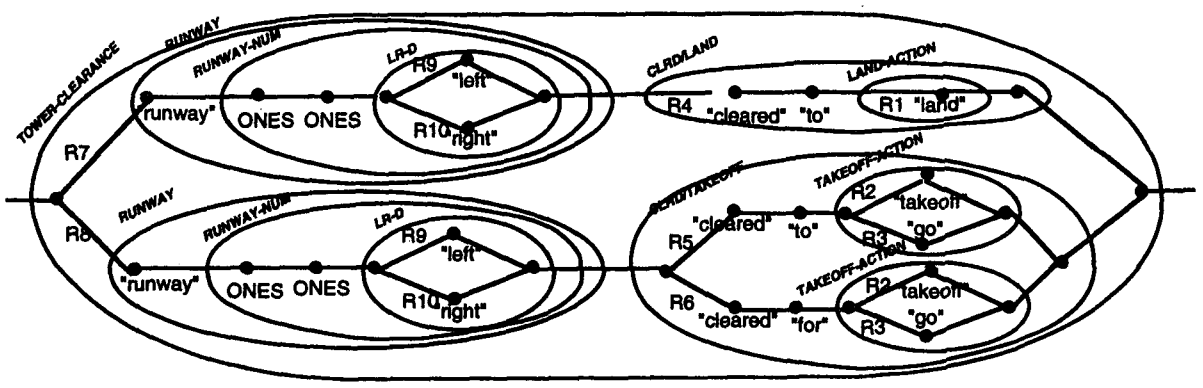

Figure 6: Finite state network

N-gram training, only in this case, nonterminals are inserted rather than replacing phrases. Note that some of the categories merely emphasize something already available from a lexical item, such as "takeoff" and "takeoff-action", whereas others capture information that is only implicit, such as the fact that "two two right" is a runway.

COMMERCIAL-AIRPLANE nera thirty four ninety eight
TURN-ORDER turn right heading two seven zero
CLRD/TAKEOFF RUNWAY two two right CLRD/TAKEOFF
cleared for TAKEOFF-ACTION takeoff

In a sense, the parser provided equivalence classes for phrases, since, for example, the nonterminal RUNWAY was added when any one of the several runways were mentioned.

As in the information extraction component, we used the parser to determine the correct scenario based on the transcripts, which in this case provided training for the model. Remember, the uncertainty in this problem is introduced by the poor recognition results; the domain is sufficiently narrow that scenarios can be classified deterministically. For each dialog (which we determine using the speaker and hearer fields in the transcript), the system parsed transmissions until an unambiguous command is found (for example "cleared to land" and "contact ground" are only given when a plane is landing), then marked all the transmissions in that dialog as to the scenario. There will be some cases that are uncertain, for example, if only part of the transcription is available, and these cases are marked "unknown" and presented to the user who may be able to find some more subtle clue to the scenario.

2.4 Event Spotting

We are also applying these techniques in other applications. In particular, we have recently performed experiments in Event Spotting, which is an extension of word spotting where the goal is to determine the location of phrases, rather than single keywords. We used the parser/extraction portion of the system to find examples of phrase types in the corpus and to evaluate the results, as well as in the language model of the recognizer. In an experiment detecting time and date phrases in the Switchboard corpus (which is conversational telephone quality data), we saw an increase in detection rate over strictly bi-gram or phoneme loop language models (Jeanrenaud, et al. 1994).

from the recognizer and the annotated transcription; testing is just on the 1st best recognition output.

3. CONCLUSION

Combining symbolic and statistical techniques in our work so far has increased both the competence and performance of our systems. We are also beginning to combine these techniques into tools to help us better understand the problems, ranging from corpus based techniques, which begin with rules and apply them to large bodies of data to find examples of specific kinds of phenomena, to statistical techniques, such as mutual information, to help us understand what features contribute the most in a probabilistic model. In the full paper, we will expand both on this aspect of our work and project forward from our experience to help assess where to best apply these methodologies.

Acknowledgments

This work was supported by ARPA and the Air Force Rome Laboratory under contract F30602-89-C-0170 and the Department of Defense.

REFERENCES

Breiman, L., Friedman, J.H., Oshen, R.A. & Stone, C.J. (1984) Classification and Regression Trees. Wadsworth International Group. Belmont, CA.

Jeanrenaud, P., Siu, M., Rohlicek, R., M., Meteer, Gish, H. (1994) "Spotting Events in Continuous Speech" in Proceedings of International Conference of Acoustics, Speech, and Signal Processing (ICASSP), April 1994, Adelaide, Australia.

Kanal and Chandrasekan (1972) "On Linguistic, Statistical and Mixed Models for Pattern Recognition" in Proceedings of the International Conference on Frontiers of Pattern Recognition, S. Watanabc (ed) Academic Press. p.161-185.

Mark, K., Miller, M., Grenander, U., & Abney, S. (1992) "Parameter Estimation for Constrained Context-free Language Models" in Proceedings of the Speech and Natural Language Workshop, February, 1992, Morgan Kaufmann, San Mateo, CA, p. 146-149.

McDonald, David D. (1992) "An Efficient Chart-based Algorithm for Partial Parsing of Unrestricted Texts" in Proceedings of the 3rd Conference on Applied Natural Language Processing, April 1-3, 1992, Trento, Italy, pp.193-200.

Meteer, M. & Rohlicek, R. (1993) "Statistical Language Modeling Combining N-gram and Context-free Grammars" in Proceedings of International Conference of Acoustics, Speech, and Signal Processing (ICASSP).

Meteer, M. & Rohlicek, R. (1994) "Integrated Techniques for Phrase Extraction from Speech" in Proceedings for

the ARPA Workshop on Human Language Technology, March 8-11, Princeton, New Jersey.

Pereira, F. & Schabes, E. (1992) "Inside-Outside Reestimation from Partially Bracketed Corpora" in Proceedings of the Speech and Natural Language Workshop, February, 1992, Morgan Kaufmann, San Mateo, CA, p. 122-127.

Rohlicek, R., Ayuso, D., Bates, M., Bobrow, R., Boulanger, A., Gish, H., Jeanrenaud, M., Meteer, M., Siu, M. (1992) "Gisting Conversational Speech" in Proceedings of International Conference of Acoustics, Speech, and Signal Processing (ICASSP), Mar. 23-26, 1992, Vol.2, pp. 113-116.

Weischedel, R., Meteer, M., and Schwartz, R. (1993) "Coping with Ambiguity and Unknown Words Through Probabilistic Models" Computational Linguistics , 19(2), pp.359-382.