Using Dialog-Level Knowledge Sources to Improve Speech Recognition

Alexander G. Hauptmann, Sheryl R. Young and Wayne H. Ward
Computer Science Department, Carnegie Mellon University
Pittsburgh, PA 15213

Abstract

We motivate and describe an implementation of the MINDS[1] speech recognition system. MINDS uses knowledge of dialog structures, user goals and focus in a problem solving situation. The knowledge is combined to form predictions which translate into dynamically generated semantic network grammars. An experiment evaluated recognition accuracy given different levels of knowledge as constraints. Our results show that speech recognition accuracy improves dramatically, when the maximally constrained dynamic network grammar is used to process the speech input signal.

1. Introduction: The Need to Integrate Speech and Natural Language

For many years, speech recognition efforts have focused on recognizing individual sentences. Natural language processing research has always assumed its input consists of a typed representation of text, with perhaps some typing mistakes. The work done on dialogs, user goals and focus for typed natural language has never been applied to speech. This is surprising since current speech technology is far from perfect and could benefit from more knowledge of constraints.

The main problem in speech recognition is the enormous complexity involved in analyzing speech input. The value of a reduced search space and stronger constraints is well known in the speech recognition community [Kimball *et. al.* 86]. To illustrate the complexity, consider that the ANGEL speech recognition system at CMU [Adams and Bisiani 86], currently generates several hundred word candidates for every word actually spoken. When processing an utterance, many choices need to be evaluated and assigned a likelihood. Reducing the search to only the most promising word candidates by pruning often erroneously eliminates the correct path. By applying knowledge-based constraints as early as possible, one can trim the exponential explosion of the search space to a more manageable size without eliminating correct choices.

To demonstrate a new approach in speech recognition, we have built **MINDS**, a Multi-modal, INteractive Dialog System. It allows a user to speak, type and point during a problem solving session with the system. MINDS works in a resource management domain, featuring ships deployed by the navy. The basic problem situation involves a damaged ship performing a task, which needs to be replaced by a different ship with similar attributes. The solution should have minimal impact on other mission operations. For the purposes of this paper, MINDS can be viewed as a speaker-independent continuous speech recognition system that uses dialog knowledge, user goals and focus to understand what was said in its naval logistics problem solving domain. The system uses this higher level knowledge of dialogs and users to predict what the current user will talk about next. The predictions drastically reduce the search space before the sentence and word detection modules even begin to analyze the speech input.

1.1. Focus, Dialogs, Goals, and Problem-Solving Strategies

There has been much research on dialog, discourse, focus, goals and problem solving strategies in the natural language processing community. We will only briefly mention the key issues which influenced the design of the MINDS system.

Grosz [Grosz 77] found that natural language communication is highly structured at the level of dialogs and problem solving. She showed how the notion of a user focus in problem solving dialogs is related to a partitioning of the semantic space. Focus can also provide an indication how to disambiguate certain input. Additional work by Sidner [Sidner 81] confirmed the use of focus as a powerful notion in natural language understanding. She used focus to restrict the possibilities of referent determination in pronominal anaphora.

Schank and Abelson [Schank and Abelson 77] point out the power of scripts in representing and predicting sequences of events. While they applied their scripts to stories, it is clear that the same mechanism can be applied to dialog and discourse, as Robinson [Robinson 86] demonstrated.

Newell and Simon [Newell and Simon 72] were key influences in the study human problem solving. Among other things, they showed how people constantly break goals into subgoals when solving problems. Their findings, as well as much of the other research done in this area [Litman and Allen 87] illustrate the function of user goals represented as goal trees, and traversal procedures for goal trees.

1.2. Current Speech Recognition Research

The speech recognition literature shows several different approaches to limiting the search space. We will only review how other speech systems apply constraints to sentences, dialogs and user goals. Surprisingly, almost none of them use dialogs, user goals or user focus to aid speech recognition.

[1]We wish to acknowledge Ed Smith and Philip Werner. This research would not have been possible without their assistance. This research was sponsored by the Defense Advance Research Projects Agency (DOD), ARPA Order No. 4976, monitored by the Air Force Avionics laboratory under contract F33615-84-K-1520. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Defense Advanced Research Projects Agency or the US Government.

One approach to increasing constraints and reducing search space uses Markov modelling techniques [Bahl *et. al.* 83, Stern *et al.* 87, Ward *et al.* 88]. These systems rely on empirically derived transition probabilities between words to process the input. The systems are trained on large amounts of data, where conditional transition probabilities are computed between pairs or triplets of words, also known as bigram or trigram grammars. There is no notion of dialog structure, focus of attention or goals incorporated into the transition probabilities.

Several speech recognition systems claim to have dialog, discourse or pragmatic components [Lea 80]. However, all of these systems only use this knowledge above the sentence level like any typed natural language system would. The input is transformed into appropriate database queries, anaphora are resolved, and elliptic utterances are completed, but the knowledge is not used to constrain the speech input processing.

The speech recognition systems which use syntactic and semantic constraints employ some form of a semantic network [Lea 80, Kimball *et. al.* 86, Borghesi *et al.* 82]. This network is the basis for a parsing module, but does not change from one utterance to the next. All reasonable constraints about the structure and content of single sentences are embedded into the networks.

Some other speech recognition systems emphasized semantic structure over syntactic constraints [Hayes *et al.* 86]. These approaches leave too much ambiguity in the syntactic combination possibilities, with poor recognition results due to lack of constraints. The level of analysis of the semantic systems also stops with single sentences. Restrictions involving several sentences in sequence are not considered.

While none of the above speech recognition systems account for constraints beyond the sentence level, two systems do use some knowledge beyond single sentences.

1.3. Speech Recognition with Dialog Knowledge

Barnett [Barnett 73] describes a speech recognition system which uses a "thematic" memory. It predicts previously mentioned content words as highly likely to re-occur. In addition, he refers to a dialog structure, which limits possible sentence structures in the different dialog states. No actual results are mentioned in this report.

Fink and Biermann [Fink and Biermann 86] implemented a system that used a "dialog" feature to correct errors made in speech recognition. Their system was strictly history based. It remembered all previously recognized meanings (i.e. deep structures) of sentences as a dialog. If the currently analyzed utterance looked similar to one of the stored sentence meanings, the stored meaning was used to correct the recognition of the new utterance. Significant improvements were found in both sentence and word error rates when a history prediction could applied. The history constraint was only applied after a word recognition module had processed the speech, in an attempt to correct possible errors.

1.4. Innovations of the MINDS System

The MINDS system represents a radical departure from the principles of most other speech recognition systems. We believe that we can exploit the knowledge about users' problem solving strategy, their goals and focus as well as the general structure of a dialog to constrain speech recognition down to the signal processing level. In contrast to other systems, we do not correct misrecognition errors after they happen, but apply our constraints as early as possible during the analysis of an utterance. Our approach uses predictions derived from the problem-solving dialog situation to limit the search space at the lower levels of speech processing. At each point in the dialog, we predict a set of concepts that may be used in the next utterance. This list of concepts is combined with a set of syntactic networks for possible sentence structures. The result is a dynamically constructed semantic network grammar, which reflects all the constraints derived from all our knowledge sources. When the parser then analyzes the spoken utterance, the dynamic network allows only a very restricted set of word choices at each point. This reduces the amount of search necessary and cuts down on the possibility of recognition errors due to ambiguity and confusion between words.

In the next sections we will describe the use of dialog and problem-solving strategy knowledge within the MINDS system in more detail. We also present results of an evaluation of the system using the different levels of knowledge.

2. Tracking Dialog, Goals and Focus

The MINDS system maintains information on what has been talked about and what is likely to be talked about next. To do this, the dialog module has information about goal trees, which describe the individual goals and subgoals at each point in a problem solving session. A goal tree contains the concepts whose values the user will need to know about to solve the problem. The goal trees are indexed to a dialog script [Schank and Abelson 77], which determines the sequences of goal trees a user could visit.

The following aspects of a dialog and goals are used by the MINDS system.

- **Dialog Phase Knowledge.** The problem-solving dialog is broken into certain phases, similar to a script. Each phase has an associated set of goal trees. These goal trees consist of domain concepts which are considered the individual goals. A goal tree is structured as an AND-OR tree. Thus, the tree defines the goals and subgoals as well as the traversal options a user has. The goal concepts can be optional or required, single use or multiple use. We expect these goal concepts to be mentioned by the user during a particular dialog phase.

  In addition to the concepts, a dialog phase also has a set of predicted syntactic sentence structures. These are in the form of recursive transition networks and specify the kinds of sentences that will occur as a user utterance.

  For example, in a dialog phase directed at assessing a ship's damage, we expect the ship's name to appear frequently in both user queries and system statements. We also expect the user to refer to the ship's capabilities. The predicted syntactic structures are questions about the features of a ship like *"Does its' sonar still work"*, *"Display the status of all radars for the Spark"* and *"What is Badger's current speed"*.

- **Restrictions of Active Concepts.** Some goal concepts which are active at a goal tree node during a particular dialog phase have been restricted by previous dialog states. These restrictions may come either from the users utterances or from the system responses. Each phase thus not only has a list of active goal concepts, but also

a list of goal concepts whose values were determined by an earlier dialog phase.

In our example, once we know which ship was damaged, we can be sure all statements in the damage assessment phase will refer to the name of that ship only.

- **Ellipsis and Anaphora.** In addition to the knowledge above, we also restrict at each dialog point, what kinds of anaphoric referents are available. The possible anaphoric referents are determined by user focus. From the current goal or subgoal state, focus selects previously mentioned dialog concepts and answers which are important a this point. These concepts are expectations of the referential content of anaphora in next utterance.

  Continuing our example, it does not make sense to refer to a ship as "it", before the ship's name has been mentioned. We also do not expect the use of anaphoric "it", if we are currently talking about several potential replacement ships.

  Elliptic utterances are predicted when we expect the user to ask about several concepts of the same type, after having seen a query for the first concept.

  If the users have just asked about the damage to the sonar equipment of a ship, and we expect them to mention the radar, we must include the expectation for an elliptic utterance about radar in our predictions.

3. Expanding Dialog Predictions into Dynamic Networks

After the dialog tracking module has identified the set of concepts which could be referred to in the next utterance, we need to expand these into possible sentence fragments. Since these *predicted concepts* are abstract representations, they must be translated into word sequences with that "conceptual meaning". For each concept, we have precompiled a set of possible surface forms, which can be used in an actual utterance. In effect, we reverse the classic understanding process by unparsing the conceptual representation into all possible word strings which can denote the concept.

In addition to the individual concepts, which usually expand into noun phrases, we also have a complete semantic network grammar that has been partitioned into subnets. A subnet defines allowable syntactic surface forms to express a particular semantic content. For example, all ways of asking for the capabilities of ships are grouped together into subnets. The semantic network is further partitioned into subnets for elliptical utterances, and subnets for anaphora. All subnets are crossindexed with each dialog phase in which they could occur. Subnets are pre-compiled for efficiency. The terminal nodes in the networks are word categories instead of words themselves, so no recompilation is necessary as new lexical items in existing categories are added to or removed from the lexicon.

The final expansion of predictions brings together the partitioned semantic networks that are currently predicted and the concepts in their surface forms. Through an extensive set of indexing, we intersect all predicted concept expressions with all the predicted semantic networks. This operation dynamically generates one combined semantic network grammar which embodies all the dialog level and sentence level constraints. This dynamic network grammar is used by the parser to process an input utterance.

To illustrate this point, let us assume that the frigate "Spark" has somehow been disabled. We expect the user to ask for its capabilities next. The dialog tracking module predicts the "shipname" concept restricted to the value "Spark" and any "ship-capabilities" concepts. Single anaphoric reference to the ship is also expected, but ellipsis is not meaningful at this point. The current damage assessment dialog phase allows queries about features of a single ship.

During the expansion of the concepts, we find the word nets such as "the ship", "this ship", "the ship's", "this ship's", "it", "its", "Spark" and "Spark's". We also find the word nets for the capabilities such as "all capabilities", "radar", "sonar", "Harpoon", "Phalanx", etc. We then intersect these with the sentential forms allowed during this dialog phase. Thus we obtain the nets for phrases like "Does it/Spark/this_ship/the_ship have Phalanx/Harpoon/radar/sonar", "What capabilities/radar/sonar does the_ship/this_ship/it/Spark have", and many more. This semantic network now represents a maximally constrained grammar at this particular point in the dialog.

4. Parsing Speech Input with Dynamic Networks

When a user speaks an utterance, the ANGEL [Adams and Bisiani 86] front-end produces a network of phonetic labels from the input signal. In principle, any front end that produces a phoneme network could be used. The left-to-right parser we have implemented takes this network of phonemes produced by the acoustic-phonetic front-end as input and forms a set of phrase hypotheses. It builds phrases by starting at the beginning of an utterance and adding words to the end of current phrase hypotheses until the end of the utterance is reached. As each phrase hypothesis is extended, only words specified in the dynamic grammar network are even considered. A lexicon contains a network of phonemes that represent allowable pronunciations of each word. These word models are generated by applying a set of rules to the base form phonemic transcription of the word [Rudnicky 87].

If sufficient evidence is found for a grammatically correct word, that word is appended to the phrase hypothesis. Phrase hypotheses are ranked according to a plausibility score which reflects the cumulative scores of the component words. These word scores in turn are based on the scores for individual phoneme matches and the overall match of the word model. A beam search is used to limit the number of possibilities, so that only phrases within a predefined range of the current best-scoring phrase are retained. Thus the parser produces a rank-ordered set of phrase hypotheses that span the utterance.

5. Speech Recognition Results

What does the integration of dialog, goals and focus knowledge buy in our speaker-independent, continuous speech recognition system? To test the effectiveness of the use of this knowledge in MINDS, 5 speakers (3 male, 2 female) spoke to the system. To assure a controlled environment for these evaluations, the subjects only spoke the sentences prepared in three sample dialog scripts, which contained 30, 21 and 10 sentences each. The three dialogs differed in the number and specificity of the questions asked. Each speaker spoke all sentences in all three dialogs. An excerpt of a dialog sequence

can be found in Figure 1.

To prevent confounding of the experiment due to misrecognized words, the system did not use its own speech recognition result to change state. Instead, after producing the speech recognition result, the system read the correct recognition from a file which contained the complete dialog script. Thus the system always changed state according to a correct analysis of the utterance.

```
The Badger is disabled.
What capabilities did it have?
What was Badger's speed?
Show me its mission area ratings.
Which frigates have harpoons?
Phalanx?
What are their other capabilities?
What is the speed of the Kirk?
What are the mission ratings for Kirk?
```

Figure 1: An excerpt of a dialog used to evaluate the MINDS system

The system was only tested with a vocabulary of 205 words, even though the complete vocabulary is 1029 words. Since we were using an older, experimental version of the ANGEL front-end [Adams and Bisiani 86], our recognition results where substantially worse than for the current official CMU speech system. However, the point we wish to make concerns the relative improvement due to our knowledge sources, not the absolute recognition performance of the total speech system. We compare two levels of constraints: using sentential knowledge constraints only and using all the power of the dialog predictions. Thus each utterance was parsed with two different levels of constraint.

- The **"sentential level"** constraints used the grammar in its most general form, without partitioning. The constraints found in the combined semantic network of all possible sentence structures were used. The network grammar was the same for all utterances in all dialogs. This only allowed recognition of syntactically and semantically correct sentences, but ignored any user goals, focus or dialog knowledge. In addition, we used all the word level constraints. These include knowledge of word pronunciation and coarticulation rules. The sentential level is the equivalent of all the knowledge employed by most existing speech systems, as discussed earlier.
- Using all **"dialog knowledge"** constraints, we applied all the knowledge built into the system at every level. In particular all applicable dialog knowledge was added to improve performance of the system. The grammar was dynamically reconstructed for each utterance, depending on the dialog situation, user focus and goals. Thus the grammar was different for almost every utterance. Of course, the word and sentential level knowledge was also used.

Table 1 shows how the three dialog scripts compare in terms of their "difficulty" for speech recognition. A standard measure of "difficulty" is the average branching factor of the grammar. This indicates how many choices the speech recognition system is faced with when trying to identify a word. Generally, a lower branching factor indicates higher constraint and better recognition because the system has fewer choices to make. This results in fewer errors in the speech recognition process. Perplexity is another related measure obtained by taking 2 raised to the power of the entropy of the grammar. The **test set branching factor** is computed by tracing the path of each utterance through the nets and averaging the actual branching possibilities encountered during a correct parse. **Test set perplexity** is the perplexity for the nodes actually traversed during a particular utterance.

Complexity of the Recognition Task		
Constraints used:	sentence	dialog
Dialog 1 Test Set B.F.	66.0	14.1
Dialog 2 Test Set B.F.	61.0	14.4
Dialog 3 Test Set B.F.	63.2	14.4
Dialog 1 Test Set Perpl.	33.0	9.0
Dialog 2 Test Set Perpl.	29.1	9.7
Dialog 3 Perpl.	32.0	10.7
Combined Test Set B.F.	63.8	14.2
Combined Test Set Perpl.	31.5	9.5

Table 1: Average test set branching factor and perplexity for the actual utterances used in the evaluation dialogs

The dialog scripts had 14.1, 14.4 and 14.4 test set branching factor and 9.0, 9.7 and 10.7 test set perplexity, respectively for the combined dialog constraints. For the sentence level constraints, the dialog scripts showed 66.0, 61.0, 63.2 as the test set branching factor and 33.0, 29.0 and 32.0 as test set perplexity. While the three dialogs show roughly equivalent difficulty for speech recognition, we see a drastic reduction in complexity from our dialog knowledge sources. The branching factor is cut to less than one fourth its unrestricted value and the perplexity measure shows a reduction by more than a factor of three.

Speech Recognition Accuracy Improvements				
Constraints Accuracy	sentence level		dialog knowledge	
	semantic	word	semantic	word
Dialog 1	31.2	43.9	58.1	66.6
Dialog 2	38.0	49.7	61.9	68.8
Dialog 3	22.0	36.3	52.0	60.1
Combined	32.1	44.6	58.4	66.3

Table 2: Recognition results are shown as percentage of words correct and percentage of sentence meanings correct for each of 3 dialog scripts and under 3 levels of constraint

Table 2 shows the actual parsing results for each dialog in each mode. Word accuracy refers to the percentage of spoken words which were recognized by the system. Semantic accuracy refers to the percentage of utterances to which the system reacted as if all words had been understood correctly. These often contained misrecognized small words, but the resulting meaning representation was correct. The dialog constraints yield a significant increase in accuracy for words from 44.6 to 66.3 percent and meanings from 32.1 to 58.4 percent overall. This increase in accuracy is also reflected in all individual dialogs. While the actual numbers are dependent on the particular recognition system used, the increased recognition accuracy due to the higher level constraints would be noticeable in any system.

6. Conclusions and Future Research

We have shown how one can apply various forms of dialog level knowledge to reduce the branching factor in a speech recognition task. An experiment demonstrated the effectiveness of this added constraint on the recognition accuracy of the speech system. Especially semantic accuracy improved due to these constraints.

For this domain, we hand-coded the dialog structures and the higher level knowledge into the system. For larger domains and even larger vocabularies, it would be desirable to automate the process of deriving the dialog structures and goal trees during interactions with initial users.

A strategy for backing off when the dialog expectations are violated should also be added to this mechanism. We are currently implementing such a procedure.

References

[Adams and Bisiani 86]
Adams, D.A. and Bisiani, R. The Carnegie-Mellon University Distributed Speech Recognition System. *Speech Technology* 3(2):14 - 23, 1986.

[Bahl *et. al.* 83]
Bahl, L.R., Jelinek, F. and Mercer, R.L., A Maximum Likelihood Approach to Continuous Speech Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 5:179 - 190, 1983.

[Barnett 73]
Barnett, J. A Vocal Data Management System. *IEEE Transactions on Audio and Electroacoustics* AU-21(3):185 - 186, June, 1973.

[Borghesi *et al.* 82]
Borghesi, L. and Favareto, C. Flexible Parsing of Discretely Uttered Sentences. In Horecky, J. (editor), *COLING-82*, pages 37 - 48. Association for Computational Linguistics, North-Holland, Prague, July, 1982.

[Fink and Biermann 86]
Fink, P.E. and Biermann, A.W. The Correction of Ill-Formed Input Using History-Based Expectation with Applications to Speech Understanding. *Computational Linguistics* 12(1):13 - 36, 1986.

[Grosz 77]
Grosz, B.J. *The Representation and Use of Focus in Dialogue Understanding*. Technical Note No. 151, SRI Stanford Research Institute, Stanford, CA, 1977.

[Hayes *et al.* 86]
Hayes, P.J., Hauptmann, A.G., Carbonell, J.G. and Tomita, M. Parsing Spoken Language: a Semantic Caseframe Approach. In *Proceedings of COLING-86*. Association for Computational Linguistics, Bonn, Germany, August, 1986.

[Kimball *et. al.* 86]
Kimball, O., Price, P., Roucos, S., Schwartz, R., Kubala, F., Chow, Y.-L., Haas, A., Krasner, M. and Makhoul, J. Recognition Performance and Grammatical Constraints. In *Proceedings of the DARPA Speech Recognition Workshop*, pages 53 - 59. Science Applications International Corporation Report Number SAIC-86/1546, 1986.

[Lea 80]
Lea, W.A. (editor). *Trends in Speech Recognition*. Prentice-Hall, Englewood Cliffs, NJ, 1980.

[Litman and Allen 87]
Litman, D.J. and Allen, J.F. A Plan Recognition Model for Subdialogues in Conversation. *Cognitive Science* 11:163 - 200, 1987.

[Newell and Simon 72]
Newell, A. and Simon, H.A. *Human Problem Solving*. New Jersey: Prentice-Hall, 1972.

[Robinson 86]
Robinson, J.J. Diagram: A Grammar for Dialogues. In Grosz, B., Jones, K.S. and Webber, B.L. (editor), *Readings in Natural Language Processing*, pages 139 - 159. Morgan Kaufmann, Los Altos, CA, 1986.

[Rudnicky 87]
Rudnicky, A.I. The Lexical Access Component of the CMU Continuous Speech Recognition System. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*. 1987.

[Schank and Abelson 77]
Schank, R.C. and Abelson, R.P. *Scripts, Goals, Plans and Understanding*. Hillside, NJ: Lawrence Erlbaum, 1977.

[Sidner 81]
Sidner, C.L. Focusing for Interpretation of Pronouns. *American Journal of Computational Linguistics* 7(4):217 - 231, October-December, 1981.

[Stern *et al.* 87]
Stern, R.M., Ward, W.H., Hauptmann, A.G. and Leon, J. Sentence Parsing with Weak Grammatical Constaints. In *ICASSP-87*, pages 380-383. IEEE, 1987.

[Ward *et al.* 88]
Ward, W.H., Hauptmann, A.G., Stern, R.M. and Chanak, T. Parsing Spoken Phrases Despite Missing Words. In *ICASSP-88*. IEEE, 1988.