Integration of Highly Accurate Speech Recognition With Natural Language Processing

R. Bradley Cope and David Kotick
Naval Air Warfare Center
Orlando FL

Natural language increases the flexibility to communicate with computers due to the inherent efficiency of the human half of the interface. Specifically, a natural language interface for processing speech provides an efficient means of communication if a human user's eyes and hands are occupied. Human-machine dialogue, whether typed or spoken, allows for humans to communicate or interact effectively with computer systems that are becoming increasingly complex to use because of their capabilities. For these reasons, a user interface that can process natural language has the potential for simplifying an overly complex and unfriendly working environment.

Natural Language Processing (NLP) effectively builds meaningful sentences from the basic semantic language building blocks -noun and verb phrases [1]. Further, semantic and pragmatic context of a given group of messages can be restricted in terms of specific domain information relating to events. This contextual information is then analyzed in terms of the restricted set of possible meanings the sentence may have within the given situation. Current speech recognition systems rely on a constrained syntax whereas the context based NLP relinquishes the bounding syntax. Combining a highly accurate speech recognizer and natural language processing harnesses the capabilities of both components. This paper discusses the experimentation and integration of speech recognition and natural language components leading to real-time, continuous, context correct and unconstrained speech recognition for training applications.

ABOUT THE AUTHORS

Brad Cope is an Electronics Engineer with the Simulation & Models Division of the Naval Air Warfare Center Training Systems Division. He is currently responsible for exploratory and applied speech recognition research for Navy training. His undergraduate degree is a BSEE from Temple University (5/83) and he holds a Masters in Engineering Science from Penn State University (5/91). He has worked at NAWCTSD since September 1993. Brad was previously responsible for applied research in advanced flight control and electromagnetic effects at the Naval Air Development Center in Warminster PA.

David Kotick is an Electronics Engineer with the Simulation & Models Division of the Naval Air Warfare Center Training Systems Division. His principal responsibilities include the integration, evaluation and design of microcomputer based board-level products. He has received both a BSE and MSE in Electrical Engineering from the University of Central Florida. He has worked in the area of real-time multi-processing at NAWCTSD since 1983. Recent work includes speech system syntax tool development, speech post-processing algorithms and word-scoring methods in simulation systems incorporating speech technology.

Integration of Highly Accurate Speech Recognition With Natural Language Processing

R. Bradley Cope and David Kotick
Naval Air Warfare Center
Orlando FL

INTRODUCTION

The challenge of accomplishing speech understanding goes far beyond just recognizing sounds. While it is important to accurately recognize sounds and words, it is more important to understand the intended message. Natural language processing must consider a large amount of non- linguistic, dynamically changing, contextual knowledge[2]. The technical objective of this project is to quantify the system accuracy rates of an integrated natural language processor (NLP) and robust real-time speech recognition processor under constraints of real-time throughput. Novel accuracy enhancing recognition techniques are combined with NLP to experimentally determine accuracy rates and feasibility of application to military training.

Current speech recognition technology used in training systems requires a student to spend valuable training time "teaching" the speech recognition subsystem to "recognize" his or her speech. Although there are less accurate "speaker independent" speech recognition systems which do not require this training step, neither the "speaker independent" nor the "speaker dependent" speech recognition systems are able to recognize a sufficiently large vocabulary of words to allow the user to communicate with the computer using natural language. As a consequence, the student must learn a severely constrained vocabulary before training may begin. Even when restricted to a very limited vocabulary, the percentage of words which are "misrecognized" by the computer precludes the use of natural language.

The current limitations of speech recognition technology are aggravated by factors in three areas:

(1) The size and the complexity of the vocabulary to which the speech recognition system will respond effects the performance, with the complex vocabulary typically having a higher rate of misrecognized words.

(2) Current speech recognition systems cannot emulate the ability of the human listener to correct misunderstood words based upon the semantics of the rest of the message to which they belong.

(3) A high signal to noise ratio is known to improve the speech recognition rates. However, ideal signal to noise ratios are not attainable in real environments.

A high accuracy speech recognizer integrated with a natural language processor (NLP) has the potential to accept natural language input for military training. Three potential areas of application of an efficient high level user interface are; (1) replace human role players to improve training effectiveness and repeatability, (2) construct human machine dialogue database systems to assist in the development of training environments, and (3) automatic processing of recorded language for analysis of large volumes of tactical communications. Furthermore, when considering human factors, speech recognition must be sufficiently usable. The work described in this report employs unique and innovative techniques for processing speech recognition. Successfully accomplishing these techniques could yield results that permit usability by the training community. Additionally, speech recognition and natural language processing have the potential to optimize trainer interfaces as well as post analysis of trainer communications.

Basically, there are two types of continuous speech recognizers, highly accurate speaker dependent and moderately accurate speaker independent. Continuous recognition allows the speaker to speak freely whereas non-continuous systems require a pause between words. Speaker dependent recognizers are currently more accurate because they use customized stored voice templates. Current language understanding methodology using NLP and speaker independence assumes low raw recognition accuracy. Combining highly accurate speech recognition and NLP potentially yields an informal interface with reasonable accuracy. The experimental configuration is shown in Figure 1.

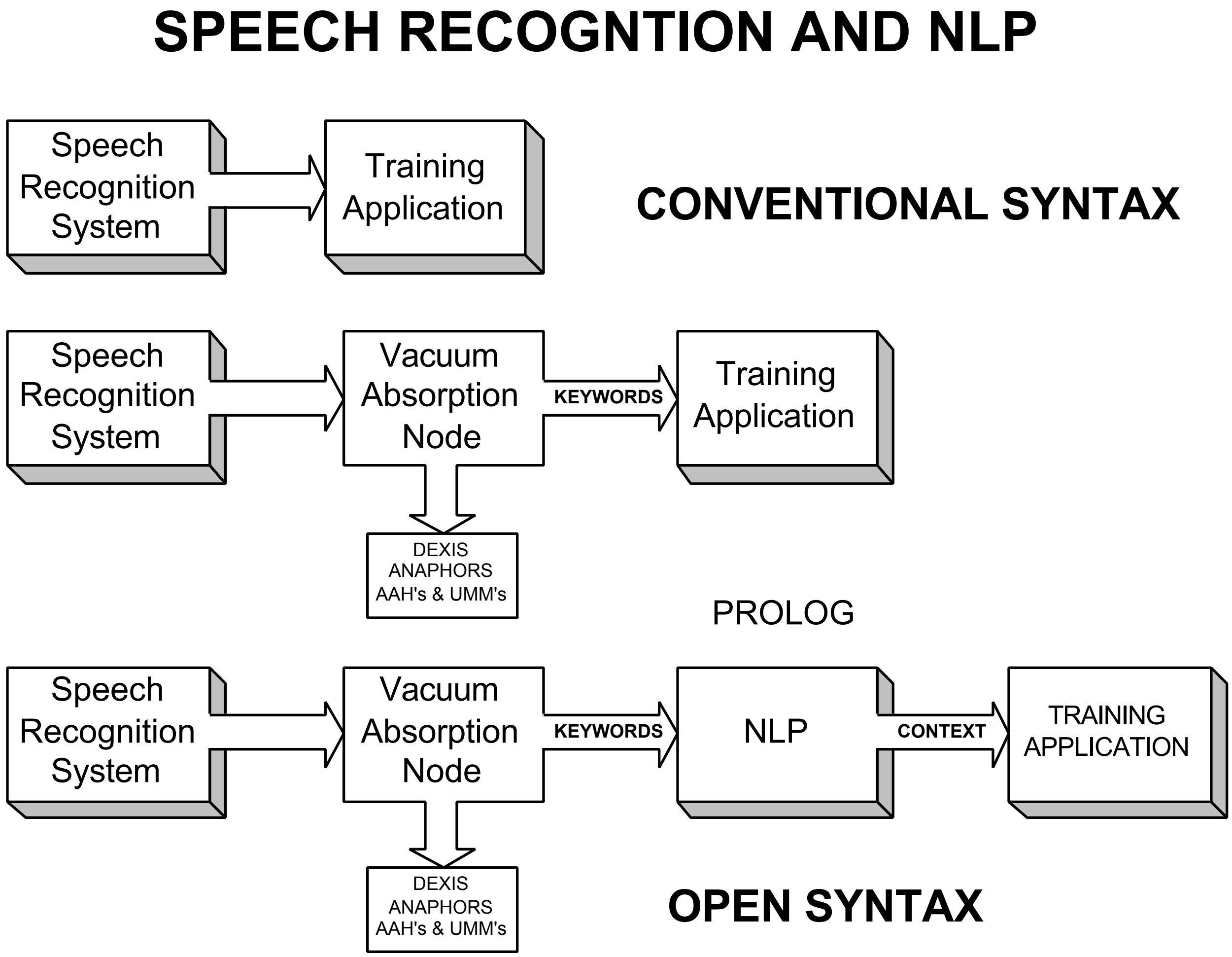

Figure 1

Generally, speech recognition systems require implementation of a user defined syntax which has the adverse effect of forcing recognition of non-domain and partial as well as valid words when a match is not found. Through experience with Dynamic Time Warp (DTW) based continuous recognizers, it has been discovered that a generic filler node can be used to vacuum or absorb some of these non-domain words thus reducing forced recognition. Additionally, known non-domain words and partials can be grouped into their own nodes effectively filtering non-essential dexis (this, these, here), anaphors (that, those, there) and prepositions as well as partial words such as “umm” and “aah”. The goal is increased recognition accuracy tailored for a natural language processor. Additionally, the combined speech recognition and NLP processes must perform in real-time to be considered for application to military training. DTW speaker dependent recognizers with a relatively simple syntax and a vocabulary of around three hundred words currently operate in real-time with a high degree of accuracy.

Speech recognizers follow a prescribed syntax where words are grouped into nodes which are then connected together to establish an optimal syntax. A limitation of these devices is that they require the user to learn the syntax. Furthermore, there is a direct relationship between the complexity of the syntax (called perplexity) and the accuracy of the real-time recognizer. Any change to the syntax due to changes in system requirements results in a change in perplexity and an exhaustive analysis to determine any adverse impact to recognition accuracy. Alternatively, NLP approaches a nearly unconstrained syntax which minimizes non-productive training time. Implementing a relatively large vocabulary with a simplistic syntax adversely affects the speech recognition accuracy, but adding an NLP post process establishes the context of what was spoken thereby potentially improving accuracy and eliminating the need to learn any structured and constrained syntax. Additionally, the NLP system can be more computationally intensive measured against a conventional system with a constrained syntax and the performance is thereby adversely affected, a tradeoff between accuracy and performance.

NLP can be used effectively to build meaningful sentences from basic semantic language building blocks -noun and verb phrases. Further, semantic and pragmatic context of a given group of messages can be restricted in terms of specific domain information relating to events. This contextual information is then analyzed in terms of the restricted set of possible meanings the sentence may have within the given situation.

Natural language processing techniques, such as parsing the parts of a communication to interpret the syntactic meaning are applied recursively to speech recognition system output in order to correct misunderstood parts of the communiqué. Further, the natural language processor is computationally intensive, but as memory and computers become cheaper and faster respectively, it can be expected that coupling of logic to speech recognition would be possible. Further, large vocabulary speech recognition systems such as Carnegie Melon University's Sphinx-II speech system runs at two times real time on a P6 with 128 MB of memory. Observing that the rate of increased processor instructions per second approximately doubles every year, systems such as the Sphinx-II will soon operate in real time. Similarly, implementation of NLP in real time will be possible because NLP is typically written in Prolog which works well with parallel processors because Prolog works in parallel. The payoff of this effort is a measure of, continuous, context correct and unconstrained speech recognition for training applications.

APPROACH

Control Model

Naturally, a control model was created as a baseline to measure performance. This model was constructed to optimally operate with a high level of accuracy (>95%). The vocabulary is relatively small, but reasonably complex to represent the level of difficulty implemented in a typical training device using speech recognition. The syntax was created to best replicate an unconstrained grammar by limiting the realm of operation to a simple application and inserting many synonyms. The vocabulary was established by interviewing 12 test subjects and verbally playing the game of tic-tac-toe. The test subject was asked to speak the intended move while the investigator scribed the resultant entry into the game grid. Responses were recorded and collated to form a comprehensive vocabulary. Subsequently, the system was evaluated for accuracy and performance. Figure 2 shows an optimized architecture of the control syntax where each node represents a word or group of words that can logically precede or follow another node. The words are not shown for simplicity.

SYNTAX FOR THE CONTROL MODEL

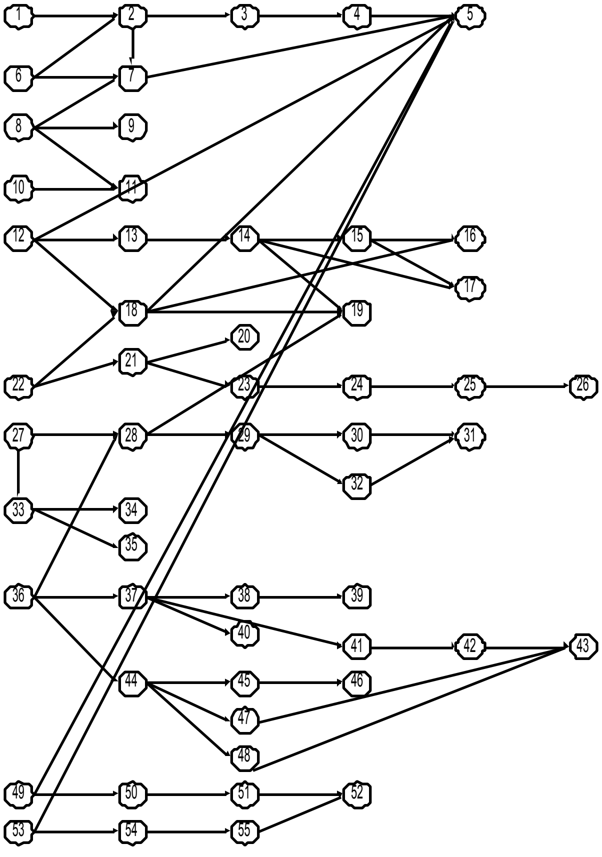

FIGURE 2

Repeatability of the spoken input is achieved by storing a digitized sample which is kept constant throughout the experimentation. In this way, measurement of the performance is subject to the integration of the experimental methodologies.

Integration

Conventional speech recognition systems require a structured syntax to limit the number of possible search paths for a given utterance. Typically an application is created for a group of users who share a common phraseology which reasonably bounds the vocabulary. The approach employed for this effort limits the number of possible nodes in the speech recognizer by using a prolog post process to determine valid relationships. The NLP prolog system is created around the vocabulary to limit the context of what's been said to relative utterances. Additionally, using a generic bucket node to vacuum or absorb non-domain words and bits of non-sensical speech significantly limits the possible number of search paths that the prolog would have to go through if the bucket node were not there. Implementing this approach significantly simplifies the development of speech recognition systems for training applications because the implementor can create the application in terms of logic relationships rather than exact structured utterances. Most importantly, the input to the system is unstructured freely spoken speech, that is, the operator can speak informally and is no longer required to learn a syntax.

Natural language processing is used to effectively build meaningful sentences from the basic semantic language building blocks such as noun and verb phrases. Similarly, semantic and pragmatic context of a given group of messages is restricted in terms of specific domain information relating to events. Prolog is especially well suited to accomplish pragmatic context relationships because of backtracking where relationships are tested recursively to apply newly satisfied matches to previously examined relationships. Using a model of logic, prolog relates to how human beings think and is conveniently machine independent. Employing prolog frees the implementer from programming in terms of computer processes and allows the effort to be focused more on the logic of the problem.

Basic grammar is a familiar implementation of pragmatic processing which can be used to validate sentence structure. Coupling the check for grammar and limiting recognized vocabulary to the target phraseology would appear to provide a reasonable solution for determining the correctness of what's been said, but configuring a speech recognizer to operate with a natural language processor significantly increases the number of possible relationships the recognizer must examine. Further, just checking for grammar would not be effective because of the reduced accuracy of the speech recognizer. Inherently, speech recognition systems effectively have fixed and limited context relationships established by the syntax. Reducing the number of nodes in a syntax by grouping possible utterances into large nodes significantly increases the number of context relationships which significantly increases perplexity. Therefore, more elements per node can reduce the speech recognizer's accuracy because elements within each node compete against each other. One can deduce that a test for basic grammar in this case would not necessarily provide a conclusive solution to the recognizer with reduced accuracy because of the increased possibility of irrelevant words. A hybrid pragmatic and basic grammar solution is needed which checks for nouns and verbs as well as context within a situation. Determining the context of the spoken input is accomplished by porting the output of the recognizer to a natural language processor which recursively examines the incoming word stream for nouns and verbs until a match for relation to the current context is found.

Integration of a prolog based natural language processor significantly simplifies the architecture of the syntax resident in the speech recognizer because the NLP is searching for context. Specifically, prolog is inherently an unconstrained syntax which searches all possible relationships for a match. Within a situation, only a subset of the possible relationships are valid. Only the utterances that relate to the current situation are examined. This significantly reduces the computational demand on the system as well as the time required for a solution. Figure 3 shows the simplified syntax used with natural language processing.

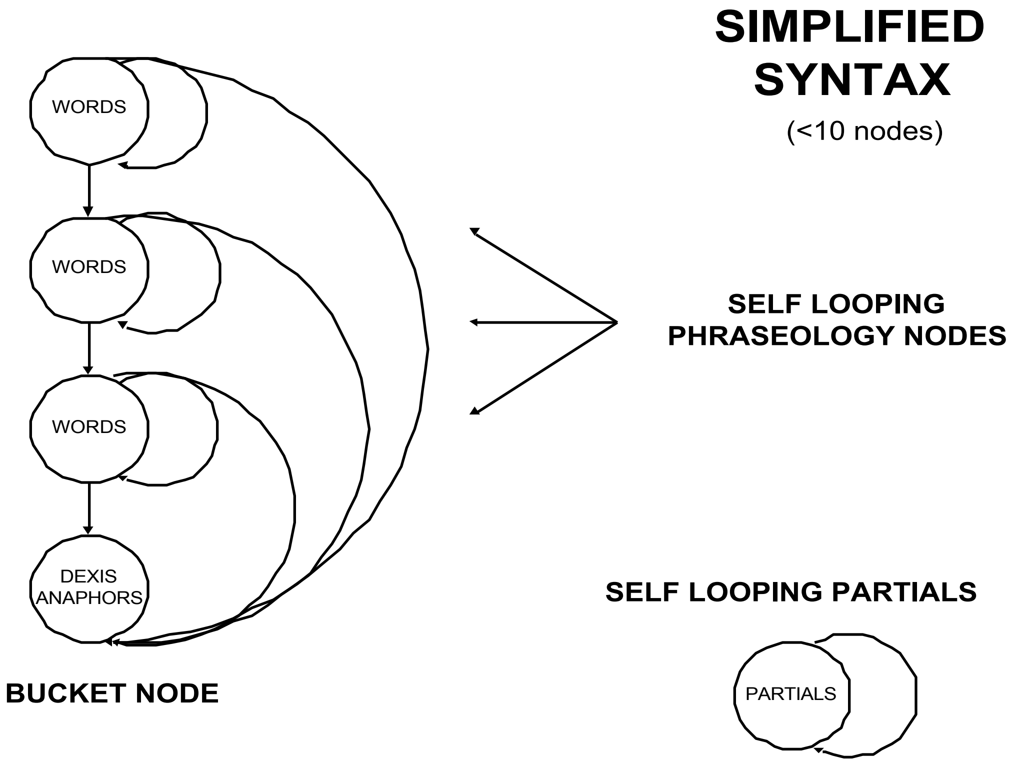

FIGURE 3

Comparing this syntax with the control model syntax in figure 2, it is clear that the control model syntax is highly structured and complex as well as difficult to modify. Looking at the simplified syntax coupled with natural language processing, it can be understood that establishing relationships describing how the words affect the functionality of the system now resides within the natural language component whereas conventional systems rely on the formal establishment of a fixed protocol. Additionally, implementation of the underlying logic for the application is enacted using methods similar to how humans think. Figure 4 shows the integrated system. Table 1 compares the perplexity, word & phrase accuracy as well as the context correctness of the two implementations. It can be seen that the measure of perplexity, which includes the number of following words per node, is a factor of 12 greater for the system configured for the natural language processor. The effect is a 7% degradation of word accuracy for an eight sample test with two test subjects. Looking at the broad picture, a study [3] of an operational system using a strict phraseology shows that 16% of the spoken input was out of phraseology. Subtracting this 16% of words spoken out of phraseology to the word accuracy rate for the conventional system yields a resultant word accuracy of only 80% indicating that the NLP syntax may be more effective for a small vocabulary. Integration of the natural language processor will mitigate the loss in word accuracy as well as words spoken out of phraseology, but will slow the system down considerably.

	Perplexity	**Word Accuracy**	**Phrase Accuracy**	**Context Accuracy**
Conventional	**4.6**	**96%**	**77%**	**99%**
***Conventional**	**4.6**	**80%**	**71%**	**83%**
NLP syntax	**54.4**	**89%**	**63%**	**89%**

*Shows impact of words spoken out of phraseology or out of order

Table 1

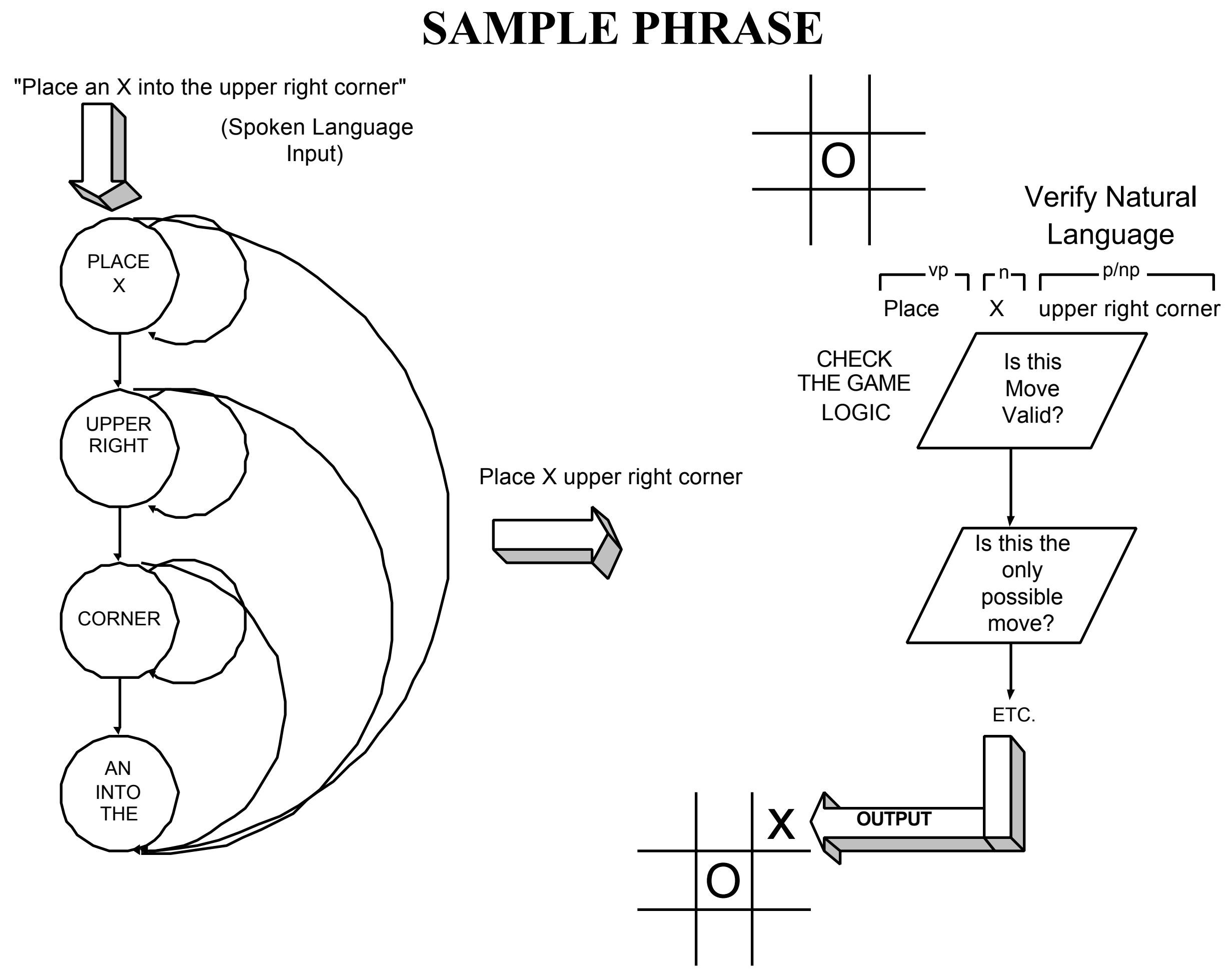

FIGURE 4

There are methodologies whereby models are established for two and three word combinations called biphones and triphones[4]. Grouping of words in this way reduces the number of possible relationships that would be searched otherwise. Search algorithms such as the Viterbi[5] beam search determine the best possible search path. The search path is based on probabilities between possible states. Integration of pragmatic solutions based on prolog processing could improve the probability of finding the correct state by weighting the transition probabilities between states based on context. The Naval Research Laboratory arrived at a similar conclusion at the close of the Interfis experiment[6]. This is a great topic for future research.

Semantics

A paper on natural language computing would not be complete without discussing semantics. Semantics, or meaning, is where language connects with the real world. This is where the rubber meets the road. Humans are very good at word-sense disambiguation such as inferring the correct meaning of the previous sentence. Additionally, just looking at keywords is not conclusive such as "*John loves Mary"* is not necessarily interchangeable with *"Mary loves John"*. Further, some statements are implicit such as "*can you close the door?"* probably means "please close the door". Prolog allows for large, complex data structures that are easy to build and modify which can represent syntactic and semantic structures[7], therefore, word-sense disambiguation can be accomplished if the relationships are defined.

Filtering

Speech recognition system typically employ a dual element noise canceling microphone. Sound that enters both microphones is canceled by diametrically combining the signals. Only sound that independently enters the primary microphone is forwarded to the recognizer. The amplitude of the incident speech to the primary microphone is significantly larger than any reflected input to the secondary microphone. Ideally there wouldn't be reflected input. Typically, operators are seated in front of a computer workstation with a smooth monitor surface thereby reflecting some speech input the secondary microphone. Research continues in the areas of noise modeling and phased array microphones.

CONCLUSIONS

1. Preliminary results indicate that an unconstrained syntax is possible using a natural language processor.

2. The context correctness of the unconstrained test system was reduced by 9% over that of the conventional system.

3. Configuring a speech recognition system for use with a natural language processor allows for informal speech and simplifies implementation, but, can adversely affect the accuracy of the speech recognition component.

4. Words spoken out of order or out of phraseology must be considered in the accuracy measurement of a conventional system (typically 16%).

5. Implementing a vacuum absorption or bucket node significantly reduces the number of relationships examined by the natural language processor.

6. Prolog allows for large, complex data structures that are easy to build and modify which can be used for word-sense disambiguation.

RECOMMENDATIONS

1. Investigate the integration of pragmatic solutions based on prolog processing which could improve the probability of finding the correct state by weighting the transition probabilities between states based on context.

2. Continue investigation of noise canceling algorithms and phased array microphones.

3. Optimize the vacuum absorption node by experimenting with a variety of partial sounds.

REFERENCES

1. Dougherty, Lawrence *"Natural Language Computing"* by Ray C. Earlbaum Assoc., Hillside NJ., 1994

2. Grossman,Wendy M."Living Language", PERSONAL COMPUTER WORLD, pp 480-484, with permission from Canon Research Centre Europe Ltd, January 1995

3. Cope, R. Bradley, Kotick, David *"Real Time Noise Tolerant Speech Recognition"* Proceedings of the 1995 I/ITSEC

4. Ravishankar, Mosur K. *"Efficient Algorithms for Speech Recognition"*, Carnegie Mellon University, May 15, 1996

5. Viterbi, A.J. "Error Bounds for Convolution Codes and Asymptotically Optimum Decoding Algorithm" IEEE Transactions on Information Theory, vol IT-13, April 1967, pp260-269.

6. Everett, S., Wauchope, K., & Peranowski, D.*"Talking to InterFIS"* InterFIS- Natural Language Interface to the Fault Isolation Shell (FIS), NRL, Sept 1992, report# NRL/FR/5510-92-9515.

7. Covengton, Michael A. *"Natural Language Processing for Prolog Programmers"*, Prentice Hall, 1994