the recognition processes," in *Proc. 3rd Int. Joint Conf. Artificial Intelligence*, Stanford, Calif., Aug. 1973, pp. 185–183.
[18] R. Reddy and A. Newell, "Knowledge and its representation in a speech understanding system," in *Knowledge and Cognition*, L. W. Gregg, Ed. Washington, D. C.: Erlbaum, ch. 10, to be published.
[19] E. Rich, "Inference and use of simple predictive grammars," in *Proc. IEEE Symp. Speech Recognition*, Carnegie-Mellon Univ., Pittsburgh, Pa., Apr. 1974, p. 242.
[20] H. B. Ritea, "A voice-controlled data management system," in *Proc. IEEE Symp. Speech Recognition*, Carnegie-Mellon Univ., Pittsburgh, Pa., Apr. 1974, pp. 28–31.
[21] P. Rovner, B. Nash-Webber, and W. Woods, "Control concepts in a speech understanding system," in *Proc. IEEE Symp. Speech Recognition*, Carnegie-Mellon Univ., Pittsburgh, Pa., Apr. 1974, pp. 267–272; also this issue, pp. 136–140.
[22] J. F. Rulifson *et al.*, "QA4: a procedural calculus for intuitive reasoning," AI Center, Stanford Res. Inst., Menlo Park, Calif., Tech. Note 73, 1973.
[23] L. Shockey and L. D. Erman, "Sub-lexical levels in the Hearsay II speech understanding system," in *Proc. IEEE Symp. Speech Recognition*, Carnegie-Mellon Univ., Pittsburgh, Pa., Apr. 1974, pp. 208–210.
[24] W. A. Woods, "Motivation and overview of BBN SPEECHLIS: an experimental prototype for speech understanding research," in *Proc. IEEE Symp. Speech Recognition*, Carnegie-Mellon Univ., Pittsburgh, Pa., Apr. 1974, p. 1–10; also this issue, pp. 2–10.

The DRAGON System—An Overview

JAMES K. BAKER

Abstract—**This paper briefly describes the major features of the DRAGON speech understanding system. DRAGON makes systematic use of a general abstract model to represent each of the knowledge sources necessary for automatic recognition of continuous speech. The model—that of a probabilistic function of a Markov process—is very flexible and leads to features which allow DRAGON to function despite high error rates from individual knowledge sources. Repeated use of a simple abstract model produces a system which is simple in structure, but powerful in capabilities.**

INTRODUCTION

TO ACHIEVE reliable speech recognition it is necessary to combine information from a variety of sources [4]. In addition to the direct acoustic information, valuable sources include the vocabulary, the grammar, and the semantics of the utterance. Extracting the information from each of these sources of knowledge is a difficult task, and the need to coordinate the various pieces of information makes the task even more difficult. For the DRAGON system a general theoretical model has been adapted to represent each of the important sources of knowledge. The sources of knowledge are organized into a hierarchical system such that the integrated system is again an instance of the same general model. The availability of this general theoretical framework has greatly simplified the DRAGON speech understanding system.

The general model which is used throughout the DRAGON system is that of a probabilistic function of a Markov process [2]. In this model there are two sequences of random variables $X(1),X(2),X(3),\cdots,X(T)$, and $Y(1),Y(2),Y(3),\cdots,Y(T)$. The X's correspond to internal states which are not observed and the Y's correspond to external observations whose distributions depend on the values of the X's. For example, the X's could represent the sequence of phones in an utterance and the Y's could represent the sequence of acoustic measurements. Alternatively, the X's could be the sequence of words in an utterance and the Y's could represent the sequence of phones and modifiers as the words are actually pronounced. Changing the frame of reference again, the Y's could represent the words of the utterance and the X's could represent the underlying sequence of syntactic-semantic states.

Manuscript received March 29, 1974. This research was supported in part by the Advanced Research Projects Agency of the Department of Defense under Contract F44620-73-C-0074 and is monitored by the Air Force Office of Scientific Research. This paper was presented at the IEEE Symposium on Speech Recognition, Carnegie-Mellon University, Pittsburgh, Pa., April 15–19, 1974.

The author is with the IBM Thomas J. Watson Research Center, Yorktown Heights, N. Y. 10598.

FEATURES OF THE DRAGON SYSTEM

The major features of the DRAGON system are

1) delayed decisions;
2) generative form of model;
3) hierachical system;
4) integrated representation;
5) general theoretical framework.

The various sources of knowledge are organized into a hierarchy of probabilistic functions of Markov processes. A network is constructed to provide an integrated representation of the hierarchy. Recognition of an utterance corresponds to finding an optimum path through the network. The optimum path if found by an algorithm which, in effect, explores all possible paths in parallel [1].

Delayed Decisions

In terms of the network representation, most speech recognition algorithms search for a suboptimum path through the network. A globally optimum path would clearly be superior, but with most models it is prohibitively expensive to compute. The Markov model of the

DRAGON system permits such a globally optimum path to be found by an algorithm such that the number of computations is linear in the length of the utterance.

The Markov assumption is a prescription to include "all relevant context" in formulating the state space of the process. By considering at each point in time all possible internal states, the algorithm searches all possible paths through the network. By combining paths when and only when they come to the same state at the same time, all decisions are delayed until the full effect of all context, past and future, has been considered.

Generative Form of Model

By having an external sequence (Y) depend probabilistically on an unobserved internal sequence (X), the system allows knowledge sources to be represented in a generative form [6]. Given the sequence of syntactic-semantic states one can generate the words. Given the words one can generate the phones. Given the sequence of phones one can generate the sequence of acoustic observations. But, computationally, this hierarchy of conditional probabilities can by reversed by applying Bayes' theorem. In analyzing a specific utterance one can proceed from the known observations to the internal states which must be inferred.

Hierarchical System

The sources of knowledge are organized into a hierarchy based on the following observation: the "top" levels of a speech recognition system change state less frequently than the "bottom" levels. Thus a single syntactic-semantic state corresponds to a sequence of several words; a single word corresponds to a sequence of several phones; and a phone corresponds to a sequence of several acoustic events. The hierarchy is not absolute—for example, syntax and semantics are mixed together into a single multilevel process—but it provides a convenient means for combining the Markov process which represent the individual sources of knowledge.

Integrated Representation

A network is constructed which represents the total hierarchy of Markov processes. The process as a whole fits the same general model as the pieces—it is a probabilistic function of a Markov process. All of the "knowledge" of the system is represented in a pair of simple data structures: the transition matrix of the network and the table of conditional probabilities connecting internal states to external observations. The main program of the system is based on the general model of a probabilistic function of a Markov process. All speech-specific knowledge is represented in the data structures, not in the program.

General Theoretical Framework

Having a general theoretical structure greatly simplifies the speech recognition system. It is both easier to implement and easier to understand. Its operations can be expressed explicitly by a simple set of mathematical equations. A powerful general system is constructed by repeated use of a flexible theoretical model.

Potential Problems and Disadvantages

Delayed decisions—searching all possible paths through the network—could lead to a combinatorial explosion in the number of computations. The Markov model completely prevents this combinatorial explosion. Alternate paths are recombined at exactly the same rate that new branches are formed. The total number of computations is linear in the length of the utterance.

The integrated representation of a hierarchical system could result in an excessively large state space. Care must be exercised as to what context must be included and what can be safely ignored. Experience indicates, however, that the network representation is a compact and powerful representation and speech recognition tasks with large vocabularies can be accommodated.

Representing all knowledge as conditional probabilities does not imply any loss of power, since the probabilities can be set to zero or to one whenever appropriate. However, it does require that estimates be computed for all the probabilities in the system. Fortunately, all these probabilities are easily estimated from the frequency of occurrence of corresponding events in a set of training utterances.

GENERAL MODEL

Let the sequence $X(1),X(2),X(3),\cdots,X(T)$ be the sequence of states of a Markov process [3] with transition matrix $A = (a_{i,j})$. Let $Y(1),Y(2),Y(3),\cdots,Y(T)$ be a sequence of random variables such that, for all t, $\Pr(Y(t) = k \mid X(t-1) = i,\ X(t) = j) = b_{i,j,k}$. Use a bracket and colon notation to abbreviate sequences. Thus $X[1:T] = X(1),X(2),X(3),\cdots,X(T)$ and $Y[1:T] = Y(1),Y(2),Y(3),\cdots,Y(T)$. The assumptions of the model are that

$$\begin{aligned}\Pr(Y(t) &= y(t) \mid X[1:t] \\ &= x[1:t],\ Y[1:t-1] = y[1:t-1]) \\ &= \Pr(Y(t) = y(t) \mid \\ X(t-1) &= x(t-1),\ X(t) = x(t)) \\ &= b_{x(t-1)x(t),y(t)} \qquad (1)\end{aligned}$$

and

$$\begin{aligned}\Pr(X(t) &= x(t) \mid X[1:t-1] = x[1:t-1]) \\ &= \Pr(X(t) = x(t) \mid X(t-1) = x(t-1)) \\ &= a_{x(t-1),x(t)}. \qquad (2)\end{aligned}$$

Under these assumptions,

$$\Pr(X[1:T] = x[1:T],\ Y[1:T] = y[1:T]) = \prod_{t=1,T} a_{x(t-1),x(t)} b_{x(t-1),x(t),y(t)} \qquad (3)$$

where a special extra state $x(0)$ is introduced and $a_{x(0),j}$ and $b_{x(0),j,k}$ are defined appropriately.

It is convenient to introduce a special notation for the total probability of all partial sequences resulting in a particular state at a particular time. Let

$$\alpha(s,j) = \Pr(X(s) = j, Y[1:s] = y[1:s])$$
$$= \sum_{x[1:s-1]} \prod_{t=1,s} a_{x(t-1),x(t)} b_{x(t-1),x(t),y(t)} \quad (4)$$

where the sum is over all possible sequences $x[1:s-1]$ and $x(s) = j$. The values of α for a given s can easily be computed from the values for $s-1$. In fact,

$$\alpha(s,j) = \sum_i \alpha(s-1,i) a_{i,j} b_{i,j,y(s)}. \quad (5)$$

Conditional probabilities based on the known sequence $y[1:T]$ can be computed from the function α and a similar function computed backwards in time from the end of the sequence. For example,

$$\Pr(X(T) = j \mid Y[1:T] = y[1:T])$$
$$= \Pr(X(T) = j,$$
$$Y[1:T] = y[1:T]) / \Pr(Y[1:T] = y[1:T])$$
$$= \alpha(T,j) / \sum_i \alpha(T,i). \quad (6)$$

Each of the sources of knowledge needed for speech recognition can be represented with this general Markov framework.

REPRESENTATION OF KNOWLEDGE SOURCES

Representing Acoustic-Phonetic Knowledge

There are several choices in how to represent acoustic-phonetic knowledge. A decision must be made whether acoustic observations should be preprocessed by specialized procedures or whether the stochastic model should deal directly with the acoustic parameters. To simplify the exposition, consider just the case in which specialized preprocessing is done.

Assume that at each time t $(1 \leq t \leq T)$, an acoustic observation is made. Each such observation consists of a vector of values of a set of acoustic parameters, which in the stochastic model is represented by a vector-valued random variable $Y(t)$. There is a sequence of phones $P[1:J]$ which is produced during the time interval $1 \leq t \leq T$. Assume that the phones occupy disjoint segments of time, that is, assume there is a sequence $s_0 < s_1 < s_2 < s_3 < \cdots < s_J$ such that $P(j)$ lasts from observation $Y(s_{j-1})$ through observation $Y(s_j) - 1$. (Set $s_0 = 1$, $s_J = T$.)

Let $p[1:J]$ be the actual sequence of phones in an utterance and let $y[1:T]$ be the actual observed sequence of acoustic parameters. For convenience, also introduce a special initalization phone $p(0)$ which is assigned a special value to allow the initial probabilities to have the same form as the transition probabilities later in the sequence. Since the actual times $s_1, s_2, s_3, \cdots, s_{J-1}$ are not known, it is necessary to associate each arbitrary segment of time with some phone. For any pair of times t_1 and t_2 let $S(t_1,t_2)$ be that value of j for which the expression $(\min(s_j,t_2) - \max(s_{j-1},t_1))$ is maximized. If $t_2 \leq 1$ then set $S(t_1,t_2) = 0$.

The acoustic preprocessor tries to estimate a phonetic transcription from the acoustics alone. By looking for discontinuities or rapid changes in the acoustic parameters, the preprocessor divides the sequence $Y[1:T]$ up into K phone-like segments $Y[1:t_1-1], Y[t_1:t_2-1], Y[t_2:t_3-1], \cdots, Y[t_{K-1}:t_K]$. Then an attempt is made to classify each segment $Y[t_{j-1}:t_j - 1]$ using some form of pattern recognition procedure. Let $t_0 < t_1 < t_2 < \cdots < t_K$ be the segment boundary times as decided by the preprocessor and introduce the random variable $D(t)$ which is 1 if there exists a k such that $t_k = t$ and is 0 otherwise. Let $F(k)$ be the label assigned by the preprocessor to the segment $Y[t_{k-1}:t_k - 1]$. (For completeness, set $t_k = t_0 = 1$ for $k < 0$, and $t_k = t_K = T$ for $k > K$.)

For some pattern matching procedures it is possible to directly estimate conditional probabilities. When using such a procedure, let

$$B[p,k] = \Pr(Y[t_{k-1}:t_k - 1] = y[t_{k-1}:t_k - 1] \mid P(S(t_{k-1},t_k)) = p). \quad (7)$$

The pattern matching procedure might yield only the label $F(k)$ representing a best guess as to the underlying phone. In such a case it is necessary to estimate the conditional probabilities from statistics of performance by the pattern matcher on training data. Let $f[1:K]$ represent the actual sequence of labels generated by the pattern recognizer for the utterance being considered. Then set

$$B[p,k] = \Pr(F(k) = f(k) \mid P(S(t_{k-1},t_k)) = p), \quad (8)$$

where the conditional probability is estimated by the frequency of such events in a set of training utterances.

In addition to estimating the probability of substitutions or confusions, it is necessary to estimate the probability of the preprocessor producing either too many or too few segments. The probability of such events may be estimated from their frequency of occurrence in a set of training utterances. Let

$$E[p_1,p_2,n] =$$
$$\Pr(D(t_{k-2}) = D(t_{k-1}) = D(t_k) = 1,$$
$$D[t_{k-2}+1:t_{k-1}-1] = 0,$$
$$D[t_{k-1}+1:t_k - 1] = 0 \mid$$

$$P(S(t_{k-2},t_{k-1})) = p_1, P(S(t_{k-1},t_k)) = p_2,$$
$$S(t_{k-1},t_k) = S(t_{k-2},t_{k-1}) + n). \quad (9)$$

If the acoustic preprocessor is reliable, then $E[p_1,p_2,n]$ should be small except for $n = 1$ and should be negligible for $n > 2$. In the DRAGON system, it has arbitrarily been assumed that $E[p_1,p_2,n] = 0$ for $n > 4$. Note that $E[p_1,p_2,0]$ is undefined and meaningless unless $p_1 = p_2$.

We can now estimate the conditional probability of the sequence $Y[1:T]$ given the sequence $P[1:J]$.

$$\Pr(Y[1:T] = y[1:T] \mid P[0:J] = p[0:J])$$
$$= \sum_{n[1:K]} \prod_{k=1,K} B[p(z(k)),k]E[p(z(k-1)),p(z(k)),n(k)], \quad (10)$$

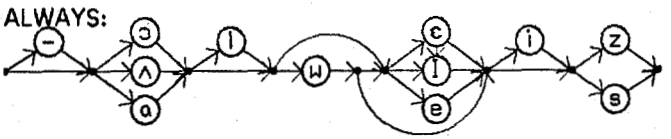
where $z(k) = \sum_{i=1,k} n(i)$ and the sum is taken over all sequences $n[1:K]$ such that $z(k) = J$. [By convention $z(0) = 0$.]

In order to apply the theory of a probabilistic function of a Markov process, it is necessary to specify the transition probabilities for the phone sequence $P[1:J]$. It is the task of the other sources of knowledge to specify these probabilities. Phonological rules may be represented either directly or indirectly in the estimates of $E[p_1,p_2,n]$ and $B[p,k]$, but all higher levels of the hierarchy deal only with the sequence $P[1:J]$ and are insolated from the acoustics $Y[1:T]$ or the labels $F[1:K]$.

Representation of Lexical Knowledge

This section discusses the computation of the conditional probability $\Pr(P[1:J] = p[1:J] \mid W[1:I] = w[1:I])$ where $W[1:I]$ is the sequence of words in the utterance and $P[1:J]$ is the sequence of phones. Knowledge of the sequence of words in an utterance is such a strong determiner of the sequence of phones that it is unusual to formulate the connection as a stochastic process. Nevertheless, the stochastic formulation can represent the same rules as other formulations and in a compact and computationally convenient form.

Let us first consider how alternate pronunciations of a particular word can be represented by a probability network. As an example, take the word "always" as used in the automatic recognition of continuous speech (ARCS), system (IBM Rockwell) [9], [5]. There are 432 pronunciations which are allowed. The ARCS system can have such a complete list of phonetic variants because it uses a network representation of the alternatives and constraints. Some speech understanding systems use an explicit list of alternate pronunciations, either generated automatically from a phonemic dictionary or preselected by hand. But an easy way to represent an exhaustive list of alternate pronunciations is by a network. The network representation for "always" is



where the dots (.) are dummy nodes introduced so that the network can be shown in two dimensions. We have represented the phones as nodes rather than as arcs (which would be even more compact) because such a representation fits more easily into the integrated system. The node-based representation permits explicit representation of sequential constraints (such as the restriction that if /w/ is omitted, then the following vowel cannot also be omitted).

The network representing alternate pronunciations of a given word can either be derived by hand and stored in a dictionary of word networks, or can be derived by automatic procedures. The automatic procedures take a canonical pronunciation and apply phonological rules to produce a network representing all likely pronunciations of the word. Even if alternate pronunciations of words are not derived by rule, the phonological rules are still important because many of them can apply across word boundaries.

The process of applying phonological rules is one way in which the DRAGON system deviates from the conceptual hierarchy. The syntax and semantics of a particular task is represented by a network in which each node corresponds to a word. Using either a dictionary of canonical pronunciations or a word-network dictionary, a small network is substituted for each word node. The result is a network in which each node is an individual phone. The phonological rules are then repeatedly applied to the network. For each phonological rule the entire network is searched to find any nodes which satisfy the context conditions of the rule. Each rule provides an alternate pronunciation of some sequence of phones. If the alternate pronunciation is not already represented then an extra branch is created in the network representing the sequence of phones for the alternate pronunciation. This process applies across word boundaries as well as within words, depending on the phonological rule. Conditional probabilities for the different branches of the phonetic network are estimated from frequency of occurrence statistics for a set of hand transcribed sentences. Such probabilities could even be made dialect dependent or even talker dependent. Note that the training sentences only need to be phonetically transcribed, it is not necessary to know the time at which each phone occurs since at this level we are no longer dealing directly with acoustics.

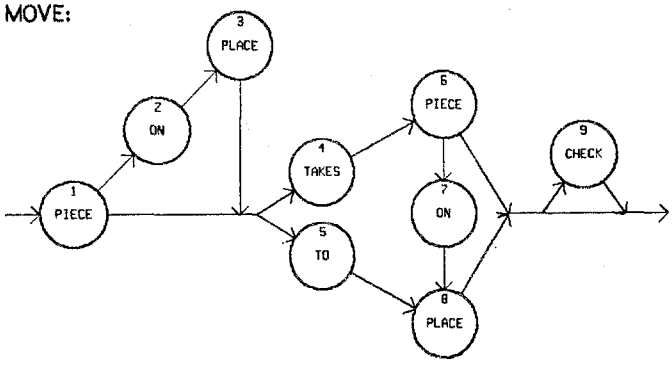
The explicit representation of phonological rules in the network is easily achieved at an expense of doubling or tripling the number of nodes in the network. However, with this stochastic network model it is not essential that an exhaustive set of phonological rules be used. In fact,

implementations of the DRAGON system have been made with no explicit phonological rules and only one canonical pronunciation for each word. The reason that this representation is possible is that any phonological phenomena which are not introduced explicitly will be treated at the acoustic-phonetic level. Thus phonological substitutions can be mimicked by adjusting the probabilities in the matrix $B[p,k]$ to include the probability that p is not the actual phone used by the talker but rather that some other phone q is spoken. Similarly, phonological insertions and deletions can be treated by adjusting the probabilities in the matrix $E[p_1,p_2,n]$. The disadvantage of this approach is that the matrices B and E represent less context than is available in the explicit representation of the phonological rules.

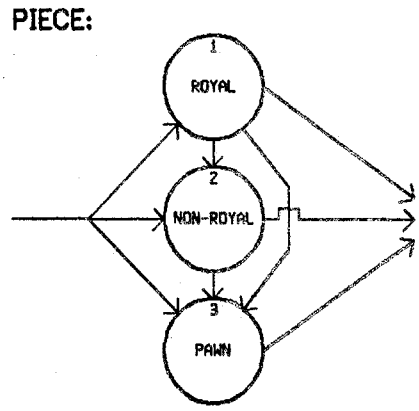
There is a serendipitous benefit in using the matrices B and E to represent acoustic-phonetic knowledge independently from the representation of the phonological rules. If the matrices B and E are estimated by running the acoustic preprocessor on a collection of test utterances, then any phonological rules which are left out in the prepared labeling of the test utterances are automatically absorbed into the estimates of B and E. Thus a perfect hand-labeled transcription of the test utterances is not only unnecessary, but undesirable. The best labeling for training purposes is an automatically generated labeling from a procedure knowing the sequence of words and having exactly the same lexical knowledge and phonological rules as the speech understanding system.

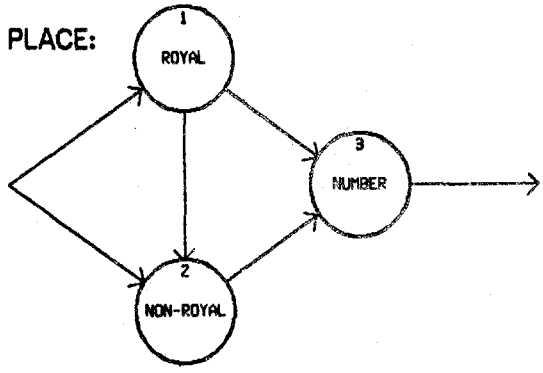
Representation of Syntactic and Semantic Knowledge

The syntax and semantics of a specific task domain can be represented by a multilevel network corresponding to a Markov process. Consider as a task a spoken chess move. Chess has a specialized grammar as well as a specialized vocabulary [6], [7]. Leaving aside a few special moves, a move can be represented by a path through the following network:

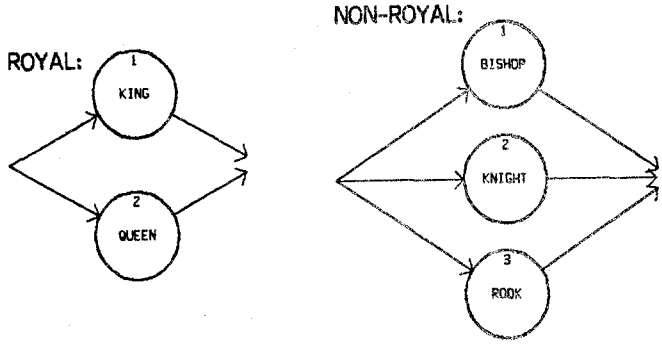


The nodes in the above network are not in general individual words, but are subgrammars which are themselves represented by networks. For example:





Again, the nodes can be expanded as networks:



It is clear that any regular (finite state) grammar can be represented by a finite network. But in a speech understanding system the distinction between a regular grammar and an arbitrary context-dependent grammar is somewhat artificial. Consider the language of utterances generated by a particular grammar, not the sequence of words but the sequence of acoustic events. It is not unreasonable to assume, for example, that each entry in $B[p,k]$ is nonzero, although perhaps very small. Such a result would automatically be the case, for example, if the conditional probability distributions for the acoustic parameters are multivariate normal distributions.

But if each entry in $B[p,k]$ is nonzero, then at the acoustic level the language must include all possible

sequences. Such a language can, of course, be represented by a finite network grammar. Thus, the issue becomes not one of generating the proper language, but rather one of modeling as accurately as possible the conditional probabilities, which can be context-dependent even for a context-free grammar. Context is represented in the network by having separate nodes for subgrammars which differ only with respect to context. For example, in the chess grammar there are two nodes marked "piece," one describing the piece which is moving and one describing a piece which is captured. There is clearly a tradeoff between the size of the state space and the amount of context which can be represented. For specialized tasks it is not difficult to achieve a reasonable representation of the grammar using most words at no more than two or three nodes. The transition probabilities for the grammar network can be estimated from statistics for a set of training sentences. A large set of training sentences should be used, but they only need to be transcribed orthographically, not phonetically, at this level of the hierarchy. If Bayesian statistics are used, the *a priori* probabilities could be set to achieve the same effect as a nonprobabilistic use of the grammar. The *a posteriori* probabilities would then be a strict improvement (as judged by the training sentences).

To the extent to which the statistics of the training sentences reflect the true probabilities for spontaneous utterances for the specific task, the probability network represents not only the syntax of the task but also all of the recognition information which can be obtained from the semantics of the available context. That is, assuming the probabilities are correct, the probability network is an optimal predictor for a given amount of context, and therefore predicts at least as well as a human who is given the same amount of context and who presumably understands the sentence (although the context in this case is not the whole sentence).

Inter-sentence semantics can also be introduced into the probability network. One way to use inter-sentence semantics is to employ a user model. Suppose there is a model for the user in a particular task which gives probabilities for the user transitioning among a finite number of states depending on the types of utterances which the user has made in the past. Conceptually this model fits in easily as an extra level in the Markov hierarchy. Computationally it requires that conditional probabilities be estimated separately for each user state. However, since the user transitions between states only between utterances, a given utterance is analyzed using only a single representation of the probability network. The probabilities in this single network are weighted averages of the probabilities for the various user states. A user model is especially valuable if certain key sentences trigger user state transitions with probability one and if for each user state a small subset of the general grammar is used. Then there is a savings in both computation and storage requirements.

PERFORMANCE RESULTS

The testing of the system is still at too preliminary a stage to make any definitive conclusions, but initial results are very promising. Simulation studies have shown that the system can perform well despite a high error rate in the acoustic preprocessor. In its first test with live speech input, the system correctly recognized every word in all nine sentences in the test.

ACKNOWLEDGMENT

The author wishes to thank L. Baum, who introduced him to the theory of a probabilistic function of a Markov process, and R. Reddy, whose encouragement, sponsorship, and personal examples have been his guiding light during this research.

REFERENCES

[1] R. E. Bellman, *Dynamic Programming.* Princeton, N. J.: Princeton University Press, 1957.
[2] L. E. Baum and J. A. Eagon, "An inequality with applications to statistical estimations for probabilistic functions of Markov processes and to a model of ecology," *Amer. Math. Soc. Bull.*, vol. 73, pp. 360–362, 1967.
[3] A. A. Markov, "Essai d'une recherche statistique sur le texte du roman 'Eugene Onegin' illustrant la liaison des epreuve en chaine," *Bull. de l'Academie Imperiale des Sciences de St. Petersbourg*, vol. VII, 1913.
[4] A. Newell *et al.*, "Speech understanding systems: final report of a study group," Comp. Sci. Dep., Carnegie-Mellon University, Pittsburgh, Pa., 1971.
[5] J. E. Paul *et al.*, "Automatic recognition of continuous speech: further development of a hierarchical strategy," Electron. Res. Div., Rockwell International, 1973.
[6] D. R. Reddy, "On the use of environmental, syntactic, and probabilistic constraints in vision and speech," Comp. Sci. Dep., Stanford University, Stanford, Calif., 1969.
[7] D. R. Reddy, L. Erman, and R. Neely, "A model and system for machine recognition of speech," Comp. Sci. Dep., Carnegie-Mellon University, Pittsburgh, Pa., 1972.
[8] D. R. Reddy *et al.*, "Working papers in speech recognition—I," Comp. Sci. Dep., Carnegie-Mellon University, Pittsburgh, Pa., 1972.
[9] C. C. Tappert *et al.*, "Automatic recognition of continuous speech utilizing dynamic segmentation, dual classification, sequential decoding, and error recovery," IBM Corp., 1971.