

## Speech Recognition in a Unification Grammar Framework

Charles Hemphill and Joseph Picone

Texas Instruments, Computer Science Center  
PO Box 655474, MS 238, Dallas Texas 75265, USA**Abstract**

Statistical signal processing approaches to speech recognition have shown great promise recently in achieving high performance in well-constrained problems. These systems typically rely upon a hierarchy of finite state automata (FSA's) to define sentence, word, and/or phone level grammars. In this paper, we describe a stochastic unification grammar system that is a generalization of the conventional Hidden Markov Model approach. Unification grammars concisely model context, providing a more powerful characterization of the acoustic data than the first order Markov process. We prove the parsing techniques required for this system by achieving the same performance with homogeneous layers of stochastic regular grammars as our best FSA-based HMM system.

**Introduction**

Statistical signal processing approaches to speech recognition have shown great promise recently in achieving high performance in well-constrained problems. High performance, speaker-independent continuous digit recognition, for example, has been successfully demonstrated in the laboratory [2] [11]. This and similar systems rely upon a hierarchy of finite state automata (FSAs) to define sentence, word, and/or phone level models [10].

Many phonetic based recognition systems today perform poorer than word based systems on small vocabulary tasks. The hypothesis for this observed degradation in performance is that word models incorporate more contextual information than low level HMM based phone models. The word models essentially have more discriminating power than the combination of phone models and a probabilistic automaton that describes words in terms of phones. The FSA's used in these systems lack the ability to effectively communicate context between the phone models.

Unification grammars (UG's) [13] have been developed to explicitly and concisely model context. Whereas both Context-Free Grammars (CFGs) and Regular Grammars (RGs) require a multiplication of rules to capture such basic linguistic phenomena as subject-verb number agreement, a single unification grammar rule suffices to capture this same agreement. UG's have become a popular formalism for natural language research [7], but have seldom been applied to the speech recognition problem. This has been caused, primarily, by the lack of probability information in the formalism itself and the inability of processors for the formalism to manipulate statistical speech information.

We introduce the notion of a stochastic unification grammar and describe a processor for the formalism. Stochastic UG's provide a framework for uniformly modeling all levels of the speech understanding problem. This includes the acoustic level, which has typically been modeled with FSA-based HMM systems. The processor correctly calculates the probability of symbols based on the probability of observations and the rule probabilities. At desired levels, the rule probabilities may be computed from training data using maximum-likelihood estimates [4].

We demonstrate the feasibility of a completely symbol-based approach by achieving the same performance with layers of stochastic regular grammars as our best FSA-based system. HMMs easily map to both stochastic RGs and FSAs, but the combined top-down and bottom-up parsing algorithm used in this system differs substantially from FSA processing techniques. Most importantly, the parsing algorithm offers computational advantages when hypotheses are needed more than once at the same time frame. This occurs frequently in large grammars and the proper treatment of this condition is essential for processing unification grammars appropriate for spoken language.

**A Stochastic Unification Grammar Framework**

In this section we introduce the concept of stochastic unification grammars and briefly outline our algorithm for processing them. The concept of chart parsing plays a prominent role in this approach, providing an efficient parsing mechanism.

**Definition:** A *stochastic unification grammar* is a four-tuple  $G_s = (V_N, V_T, P_s, S)$ , where  $V_N$  and  $V_T$  are finite sets of nonterminals and terminals,  $S \subset V_N$  is the set of start symbols, and  $P_s$  is a finite set of stochastic productions each of which is of the form  $A, p \rightarrow \alpha$ , where  $A \in V_N$ ,  $p$  is the probability of applying the rule, and  $\alpha \in (V_N \cup V_T)^*$ . Let the set of probabilities of all  $k$  stochastic productions in  $P_s$  with  $A$  on the left be  $\{p_i \mid A, p_i \rightarrow \alpha_i, i = 1, \dots, k\}$ . Then  $0 < p_i \leq 1$  and  $\sum_{i=1}^k p_i = 1$ . The nonterminals and terminals are feature-value pairs.

**Definition:** A *feature set* is a set of *feature-value pairs* of the form  $f : V$ , where  $f$  is a constant (0-ary function symbol) and  $V$  is either a constant, a variable, or a feature set. A *feature set* may be indexed with a variable using the notation  $X + FS$ . The variable may be used elsewhere to denote the occurrence of the same feature set.

Stochastic RGs and CFGs are a subset of stochastic UGs, where the symbols of the grammar must be atomic (no features are associated with them). Additionally RG rules must conform to one of the following two forms:

$$A \rightarrow w, B.$$

$$A \rightarrow w.$$

where  $B$  is a single nonterminal symbol and  $w \in V_T^*$ .

As an example of why such a formalism might be useful, consider the following simplified UG rules ( $CI$  and  $CO$  represent the input and output context, respectively):

```

digit:CI ---> digit:{in:CI, out:CO},
digit:CO.
digit:CI ---> "". % empty production

digit:C ---> eight:C. % and other digits

eight:{in:{vowel:+}, out:CO} --->
ey, eight_td:CO.
eight:{in:{vowel:-}, out:CO} --->
q, ey, eight_td:CO.

eight_td:{vowel:+} ---> dx.
eight_td:{vowel:-} ---> t.

```

These rules allow efficient expression of context between words in a digit grammar. They capture the phenomenon that a final /t/ becomes flapped when the following word begins with a vowel sound. It does this by forcing the 'in' context of the current word to agree with the 'out' context of the preceding word. For example, the digit string "88" may be pronounced as [ey dx ey t], but "83" must be pronounced as [ey t th r iy]. The same phenomenon also occurs in general English phrases such as "he ate eight apples." In a grammar involving a larger vocabulary, general inter-word context can provide additional discrimination for better recognition performance. The savings in both representation and computation in these cases can become significant.

Even in this small example, it is easy to imagine that the [ey] phone might be needed at the same time by the two different 'eight' rules. Grammars associate symbols with both observations (terminal symbols) and alternate explanations (nonterminal symbols), so that duplicate work in re-hypothesizing the same observations and partial sentence hypotheses may be avoided. This is in contrast to FSA where each candidate best path is maintained separately. Figure 1 illustrates this difference.

Chart parsing provides mechanisms for efficiently processing grammars. In stochastic chart parsing, the goal is to find the best explanation of a sentence for a given utterance by finding the best explanation of symbols that combine to compose a sentence and that adjoin at frame boundaries (thereby modeling every frame of speech data). Chart parsing derives its name from the data structure used to avoid duplicate work: the chart. The chart consists of edges that represent the current state of various grammar rules in explaining the utterance.

**Definition:** A *stochastic chart edge* is of the form  $[i, r, A, \alpha, \beta, j, p_i, p_j]$ , where  $i$  is the starting frame,  $r$  is the

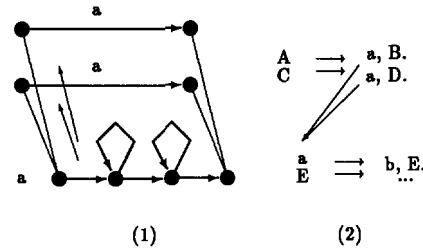


Figure 1: Processing differences between layered automata and grammars: 1) networks expanded in layered automata and 2) symbols shared in layered grammars.

production number,  $A$  is the left-hand-side of the rule,  $\alpha$  is a string of symbols that have been parsed,  $\beta$  is the remainder of the right-hand-side symbols,  $j$  is the ending frame of this edge,  $p_i$  is the initial probability, and  $p_j$  is the current probability.

The details of the chart parsing algorithm may be found in a separate paper [5]. Basically, it is similar to Earley's algorithm [3], augmented with unification [9] and probability [8], but with a delayed commitment approach to chart edge scoring [1]. This algorithm operates from left to right in a combined bottom-up and top-down fashion, providing terminal hypotheses at each time frame to lower levels and accepting completed hypotheses that began at some time in the past.

## A Hierarchy of Stochastic Regular Grammars

In this section, we describe the chart parser as applied to regular grammars. This provides an indication that the ideas are appropriate for speech processing and calibrates the system with respect to existing FSA-based HMM systems.

Figure 2 represents the data interaction in a layered stochastic regular grammar system. It can process  $n$  levels of regular grammars, allowing expansion of more than one symbol in a rule as in CFGs, but without recursive ability. The sentence grammar dictates which hypotheses propagate to lower levels at each frame. Each grammar level in turn propagates hypotheses needed in order to successfully return complete observations. The last level includes a set of grammars that represent an HMM for each phone (or word in a whole-word system). Appropriate reference data from this level is compared with the current input speech vector. The processor then incorporates the reference probabilities into the current state of the parse and any completed hypotheses pass to the next higher level as observations. Hypotheses and observations at each level propagate down and up at each frame until all of the speech data may be explained by the formation of a complete sentence.

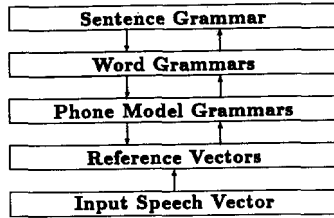


Figure 2: Grammar interaction.

It is convenient to map existing HMM systems onto the stochastic RG framework. This will also aid understanding of the system. Define a hidden Markov model by the state transition matrix  $A = [a_{ij}]_{N \times N}$  for states  $q_1, q_2, \dots, q_N$ , where

$$a_{ij} = P[q_j \text{ at } t+1 \mid q_i \text{ at } t]$$

and the symbol probability matrix  $B = [b_{jk}]_{M \times N}$  for reference patterns  $w_1, w_2, \dots, w_M$ , where

$$b_{jk} = P[\omega_k \mid q_j].$$

Assume, without generality, that each model has a single absorbing (or stop) state,  $q_{NN}$ . Then for all  $i, j$ , and  $k$ , form the following rules:

$$q_i, a_{ij} b_{jk} \rightarrow \omega_k, q_j.$$

An extra rule compensates for the fact that RGs end on observations (even if the observation is empty):

$$q_{NN}, 1.0 \rightarrow \epsilon.$$

For a given sequence of observations  $O$ , these rules are consistent with the likelihood function

$$L(O \mid A, B) = \sum_{Q^T} \prod_{t=1}^T a_{q_{t-1} q_t} b_{q_t}(O_t),$$

and allow the RG to produce identical results to that of an FSA-based HMM.

When continuous multivariate density functions are used, each  $\omega_k$  represents a reference vector and its probability is determined at each frame by a Euclidean distance calculation. In this case, the rules are of the form

$$q_i, a_{ij} \rightarrow \omega_k, q_j.$$

and the parser incorporates the probability of the reference vector into the current explanation of the rule represented by a stochastic chart edge. This separation of the probability of an observation from its environment is essential to the layered chart parsing approach: each observation is simply a symbol requiring an explanation from a given time and with a given initial probability. Note that in a stochastic UG system, context can be factored into the probability calculations, thereby producing a context sensitive distance measure.

As a practical consideration, treatment of symbols in this manner mandates a different pruning strategy. This is illustrated in Figure 3. In a stochastic chart parsing

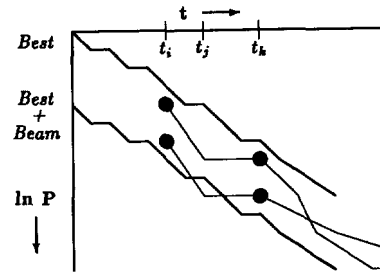


Figure 3: Effect of chart parsing on pruning.

system, the same symbol may be needed for several different explanations of the speech signal, but only the most likely representative actually becomes hypothesized. The probability of the completed observation,  $p_j - p_i$ , is then used in extending the chart edges of the remaining explanations. This leads to a situation where a lower probability explanation of the symbol may not only survive (in beam search, for example) where it otherwise would have been pruned, but the subsequent hypotheses using this symbol may actually give the more probable explanation. Since the chart parser expands only the most likely symbol, the less likely symbols at the same starting time cause no additional computation.

## Experimental Results

Two recognition experiments have been performed to calibrate the system: continuous digit strings [2] and a 1000-word Resource Management (RM) task [12]. Both systems use 18-element reference vectors with a 20 msec frame period and pooled covariance. Because of the grammars involved, chart parsing offers no advantage in the first experiment and a small advantage in the second. Table 1 indicates the relative size of the grammars.

The continuous digit experiment consists of two levels of grammars. The first grammar allows zero or more occurrences of oh, zero - nine, a silence model, and a null-speech model. The second grammar contains the HMM models for each of these, converted to RGs as indicated in the previous section. Although multiple hypotheses of the same symbol at the same time occur in the HMM grammar, the hypotheses at this level correspond to reference vectors of one frame in duration and the FSA-based system evaluates these only once.

Both systems obtain identical results when pruning is not a factor (i.e., errors occur because of the models, not pruning). Due to the differences in pruning strategies, pruning affects the results when the  $\ln$  probability beam width drops below -90 and -135 for the grammar and automaton systems, respectively. At these pruning thresholds, the ratio of the processing time of the grammar system to automaton system is 1.9.

Table 1: Grammars in the experiments.

Task	Level	Rules	Nonterms	Terminals
Digits	0	14	1	13
	1	1994	1172	517
RM	0	21025	4744	578
	1	1017	578	1017
	2	26111	13017	12001

The 1000-word resource management task consists of three levels of grammars. The first level consists of various sentence patterns desired for the task, the second level maps word types (e.g., '<ship-name>') to words. The third level contains the HMM models for the words.

Again, both systems obtain identical results when pruning is not a factor. Performance degrades below in probability beam widths of -101 and -124 for the grammar and automaton systems, respectively. At these pruning thresholds, the time ratio of the grammar to automaton system is 1.1, but bear in mind that chart parsing offers an advantage only in the level 0 sentence pattern grammar.

Table 2 summarizes the results of this section. The time ratios indicate that the overhead of chart parsing is slightly less than a factor of two, and that even for small perplexity grammars (perplexity 9 for RM), the chart parsing method begins to compensate for this overhead.

Table 2: Results of experiments.

Task	RG/FSA time	RG beam	FSA beam
Digits	1.9	-90	-135
RM	1.1	-101	-124

## Conclusions

We have demonstrated an approach to speech recognition in which the entire recognition process, including acoustic processing, consists of a hierarchy of grammars. We have shown that this approach generalizes traditional FSA-based HMM systems and that a stochastic chart parsing algorithm produces the exact same solutions as an existing FSA-based system. The shift from automata to grammars allows efficient processing of complex language models by hypothesizing symbols once per frame, no matter how many times they are needed.

As an added benefit, the chart parsing algorithm allows parallel processing of lower level hypotheses autonomously with no fundamental algorithm changes. Each level is sent a list of symbols for which it must calculate probabilities, and these lists may be split across many processors. Further, because the algorithm at each level of the system is identical, each processor executes the exact same program. This facilitates expansion of the system to arbitrarily large vocabularies and complex grammars.

The layers of regular grammars used in the experiments are completely compatible with our UG framework. Future work will focus on development of grammars that

make effective use of contextual information, and on the development of more robust distance measures that comprehend this information. Storage reclamation becomes extremely important in the UG framework and the extension of this system to spoken language systems requires a new approach to this problem. We believe, however, that the introduction of the stochastic unification grammar is a step toward a true speech understanding system.

## References

- [1] A. V. Aho and T. G. Peterson, "A Minimum Distance Error-Correcting Parser for Context-Free Languages," *SIAM Journal on Computing*, Vol. 1, No. 4, Dec. 1972.
- [2] G. R. Doddington, "Phonetically Sensitive Discriminants for Improved Speech Recognition," *Proc. ICASSP*, Glasgow, Scotland, 1989.
- [3] J. Earley, "An Efficient Context-Free Parsing Algorithm," *CACM*, Vol. 13, No. 2, 1970, pp. 94-102.
- [4] K. S. Fu, *Syntactic Pattern Recognition and Applications*, Prentice-Hall, 1982.
- [5] C. Hemphill, "A Chart Parser for Stochastic Unification Grammars," Technical Report, 1988.
- [6] S. E. Levison, "Structural Methods in Automatic Speech Recognition," *Proc. of the IEEE*, Vol. 73, No. 11, Nov. 1985.
- [7] I. Mani and C. T. Hemphill, "A Natural Language Interface for Knowledge Based Systems," *Proc. of Third Annual User-System Interface Conf.*, Austin, TX, Feb., 1988.
- [8] A. Paeseler, "Modification of Earley's Algorithm for Speech Recognition," *Proc. of NATO ASI*, Bad Windsheim, 1987.
- [9] F. C. N. Pereira, and D. H. D. Warren, "Parsing as Deduction," *Proc. of ACL*, Boston, MA, June, 1983.
- [10] J. Picone, G.R. Doddington, and J.J. Godfrey, "A Layered Grammar Approach To Speaker Independent Speech Recognition," presented at the 1988 Speech Recognition Workshop, Harriman, NY, June 1988.
- [11] J. Picone, "On Modeling Duration in Context in Speech Recognition," *Proc. ICASSP*, Glasgow, Scotland, 1989.
- [12] P. Price, W. Fisher, J. Bernstein, and D. Pallett, "The DARPA 1000-Word Resource Management Database for Continuous Speech Recognition," *Proc. ICASSP*, NY, NY, April, 1988.
- [13] S. M. Shieber, "An Introduction to Unification-Based Approaches to Grammar," *CSLI Lecture Notes*, No. 4, 1986.