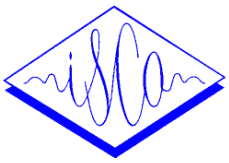

2nd International Conference on
Spoken Language Processing (ICSLP 92)
Banff, Alberta, Canada
October 12-16, 1992

BROCA, AN INTEGRATED PARSER FOR SPOKEN LANGUAGE

Tim Howells
The MITRE Corp.
Bedford, MA 01730
howells@linus.mitre.org

David Friedman
The MITRE Corp.
Bedford, MA 01730
dhf@linus.mitre.org

Mark Fanty
Oregon Graduate Institute
Beaverton, OR 97006
fanty@hobbes.cse.ogi.edu

10.21437/ICSLP.1992-77

ABSTRACT

Broca is a parser for spoken language in which natural language processing is tightly integrated with lexical and phonological processing. This is in contrast to the N-best approach usually used in speech recognition, in which the natural language component acts as an autonomous post-process to a Viterbi style search. In our system integration is achieved by expressing phonological, lexical, and natural language structures all in the form of an augmented context free grammar.

Processing in Broca proceeds through four stages. The speech signal is mapped to a perceptually based reduced representation [1]. A neural network classifier produces phoneme estimates at a frame rate of nine milliseconds. This phoneme stream is segmented using a hierarchical clustering algorithm [2]. Then the integrated grammar is dynamically matched against the segmentation by the application of a probabilistic parsing algorithm.

The advantage of a parsing approach is that it exploits higher order information present in the speech signal. This potentially includes phonological phenomena such as stress pattern and prosodic grouping, as well as the syntax and semantics of natural language. This information is lost in a Viterbi style search.

Broca is a speaker-independent, continuous speech system. We have evaluated it on an in-house database of spoken utterances for an X window manager task.

INTEGRATED PARSING

Language is characterized by hierarchical structure at several levels. One example is the familiar phrase structure tree used to describe syntactic structure. Linguists use similar trees to describe semantic content, syllable structure, prosodics, and stress pattern. These hierarchical representations can encode arbitrarily complex constraints and are a natural way to express successive levels of abstraction. Efficient parsing algorithms have been developed for the automatic recognition of such structure. These algorithms have been applied widely in the field of natural language processing, and their use has also been investigated in regards to phonological processing [3].

Broca is an "integrated" parser in the sense that a single parsing algorithm is applied to the natural language, lexical and phonological analysis. That is, the parser puts phonemes together into syllables, syllables into words, and words into analyzed utterances. There are two advantages to this approach:

- The full natural language system expressed by the parser is available to help guide and prune the search at the lower levels.
- The derived parse tree represents intermediate levels of structure (e.g. clause, syllable and sub-syllable structure) in a form that corresponds to our intuitions about language. This makes the approach more accessible to knowledge based analysis and enhancement than statistical approaches such as hidden Markov modeling.

The grammar used by Broca is a simple extension of a context free grammar for natural language. Context free grammars consist of rewrite rules that specify constituent structure. In addition to syntax rules such as $S \vdash NP\ VP$, we have rules that specify how words expand into their syllable structure, and how syllables expand into their components (onset, nucleus, and coda) down to the phoneme or subphoneme units that are the terminals in the grammar. Since the grammar will be matched against the output of a phonetic classifier, special attention must be paid to the possibility of inserted or deleted constituents.

A major problem with integrated parsing is the explosion of the search space. The Viterbi search usually used in the initial processing stage requires an amount of memory that is constant in the length of the input. The amount of memory required by a parser grows polynomially. One consequence of this is that, unlike a Viterbi search, the parser cannot work at the sample point level; there must be a prior data reduction step in which the input stream is transformed into a lattice of segments that can be used to anchor phoneme hypotheses. The way in which Broca addresses this problem will be described in the next section.

FLOW OF PROCESSING IN BROCA

Spectral Representation

The first step in processing the speech signal is to map it to a reduced representation that retains the important features for phonetic recognition. We have used a technique called perceptual linear prediction (PLP) developed by Hermansky [1]. PLP is linear prediction applied to the power spectrum after it is warped in the frequency and amplitude domains to simulate known psychophysical characteristics of human hearing. PLP has been shown to be effective in capturing speaker-independent properties of the signal in a low order representation [1, 4]. We produce these spectral vectors at a frame rate of nine milliseconds using a seventh order representation.

Phonetic Classification

We use a neural network to map the spectral vectors to a set of 39 phonetic classifications. Each category represents a phoneme, or a set of acoustically similar phonemes. A confidence level for each of the categories is produced at the nine

millisecond frame rate. The input to the neural net is the spectral vector for the frame to be classified, plus vectors from selected preceding and following frames. In our experiments we found that a large amount of context is beneficial in achieving accurate classifications [4]. The network currently used in Broca takes context from as far as 90 milliseconds on either side of the current frame.

Our network has 56 input nodes representing the current spectral vector and six context vectors. Each vector contains the seven PLP coefficients plus power. The network has 48 hidden nodes and 39 output nodes: one for each phoneme. It is trained using conjugate gradient back propagation [5]. The number of hidden nodes was arrived at empirically by optimizing test set performance given a training set of 16000 tokens [4].

Acoustic Segmentation

At each nine millisecond frame the phonetic classifier outputs a vector containing a floating point value for each phoneme. Each of these values represents a confidence level for mapping the current frame to the corresponding phoneme. This stream of vectors must be segmented into regions suitable for phoneme hypotheses. To accomplish this we have adapted an algorithm developed by Glass [2] based on hierarchical clustering. Glass applied the clustering to a stream of spectral vectors rather than vectors produced by a phonetic classifier.

The clustering proceeds by defining successively larger regions of relative acoustic homogeneity in the signal. Initially regions are defined for each frame in the utterance, and the classification vector for each frame is associated with the corresponding region. Decisions regarding merging regions are made based on a distance measure applied to their representative vectors; we use euclidean distance. Each region is compared with the largest preceding and following regions, and may be merged in the direction of greater similarity. When a merge forms a new region it is assigned a vector with elements equal to the duration-weighted average of the corresponding elements of its constituent regions' vectors. The process proceeds iteratively until the entire utterance is subsumed by a single region.

The result of this process is a binary tree of regions which is referred to as a dendrogram. Each region boundary is characterized by the distance measure at which it was merged away. The greater the distance, the greater the rate of change in the acoustic signal at that point, and the greater the likelihood that it represents a valid phoneme boundary. Due to coarticulation effects, however, even boundaries representing a fairly low rate of acoustic change must frequently be considered.

Following Zue et al. [6], we allow a phoneme hypothesis to be delimited either by a single region or by a pair of regions. This provides necessary flexibility in matching the segmentation with the phonetic grammar. The set of allowed regions (single and paired dendrogram regions) is represented in a second data structure called the acoustic phonetic (AP) network. This is the lattice of regions used by the parser to anchor its phoneme hypotheses. A significant optimization was achieved by ruling out regions for which the distance measure across either boundary is below a threshold.

As a result of the fact that we apply the region merging procedure to the neural network output vectors, the regions' representative vectors have a very useful interpretation; each value in the vector represents the average score assigned by the neural net for the corresponding phoneme over the sample points included in the region.

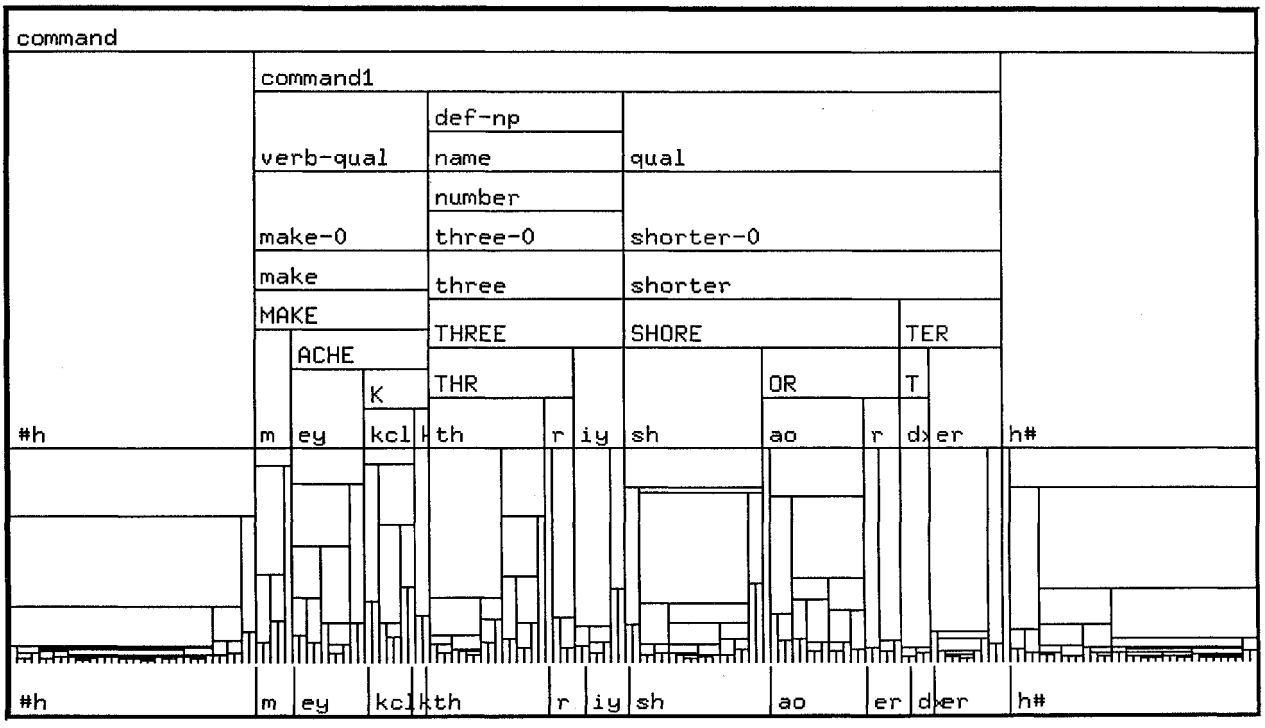

Figure 1: A parse tree derived by Broca (see text)

The Parser

Broca uses a chart parser [7] to get from the lattice of acoustic segments represented in the AP network to an interpreted utterance. The "chart" is a data structure that indexes partial interpretations by their start and end times as the parser explores the search space. The partial interpretations are called "hypotheses" or "nodes." Since the parser integrates phonological, lexical and natural language processing, a hypothesis might identify a portion of the utterance as a phoneme, a syllable, a word, a syntactic unit such as a verb phrase, and so on. The parser applies its grammar in organizing these nodes into progressively higher-level hypotheses. Terminal nodes representing phonemes are generated from the AP network regions.

Because of the uncertain nature of the input, each hypothesis is assigned a confidence level. The scores for the phoneme hypotheses are obtained from the vector associated with the AP network region delimiting the phoneme. As explained above, each value in this vector corresponds to a phoneme and represents the average score the neural net assigned that phoneme over the given region. In order to instantiate a node for the phoneme, this score is required to be above a threshold, and the region must pass duration constraints. The score threshold increases linearly with the duration of the region. This works well because it usually seems to be the case that the longer the duration, the more clearly the phoneme is articulated.

Scores propagate from constituent nodes to their parents in the same way they propagate in the dendrogram and AP network; the parent node's score is computed as the average of its constituents' scores weighted by duration. At all levels, the scores represent the average classification score obtained by the hypothesis in its full detail at the level of the nine millisecond frames. This is the metric that the parser tries to optimize.

Processing is based on the Earley algorithm [8]. The parser starts with the top level grammar symbol, traditionally called S, and tries to match it against the input starting at time zero. This causes the parser to attempt to match each possible first constituent of S as defined by the grammar, and so on in a recursive descent until it reaches terminal symbols, which correspond to phonemes. If any suitable AP network regions can be found for a required phoneme, nodes for that phoneme are entered in the chart, and the process continues with an attempt to match the next constituent of the current node or, if it is complete, of its parent. Frequently, there will be nodes or partial nodes that represent duplicate hypotheses; that is, they match the same grammar symbol to the same portion of the utterance. In these cases the node with the lower score is eliminated. When the parser pops back up to the top level with a spanning hypothesis for S, it has the best hypothesis for the entire utterance that was able to complete given the pruning thresholds that were used.

Figure 1 represents a parse tree derived by Broca for the utterance, "make three shorter." The phonetic labeling at the bottom was made by hand by an expert listening to the recorded utterance and looking at its spectogram. The dendrogram segmentation proceeds from the hand labeling up to the blocks representing phoneme labels applied by the parser. The vertical distance from the origin in the dendrogram corresponds to the distance measure at which the merges occurred. Note that we have called the top level grammar symbol *command* rather than S.

RESULTS

Broca is a speaker-independent, continuous speech system. We have tested it on a small "X windows manager" domain. This allows a person using a work station to manipulate windows via spoken commands such as "Create an Emacs window", and "Iconify the Xterm window." We recorded a database of 20 speakers each saying 30 such commands. We used 16 speakers as a training set for the neural network phonetic classifier. This was augmented with data from the TIMIT database [9]. We used the remaining four speakers and their 120 utterances as our test set. The domain has a vocabulary of 40 words, and a grammar with a perplexity of 3.2 at the word level.

The search implemented by the parser can be controlled by varying the pruning thresholds; as the thresholds are lowered, it runs more accurately but also more slowly. Figure 2 shows word accuracy as a function of processing time. The horizontal axis is labeled in multiples of real-time. These results are for a Sun SPARC 2 work station. Word accuracy levels off at 98.4% at about 4.4 times real-time. In practice we set the thresholds much higher, so that performance is near real-time. We find that people quickly adapt to the recognizer, and as a result error rates are acceptable. Broca works best on relaxed, continuous speech, so this is no hardship.

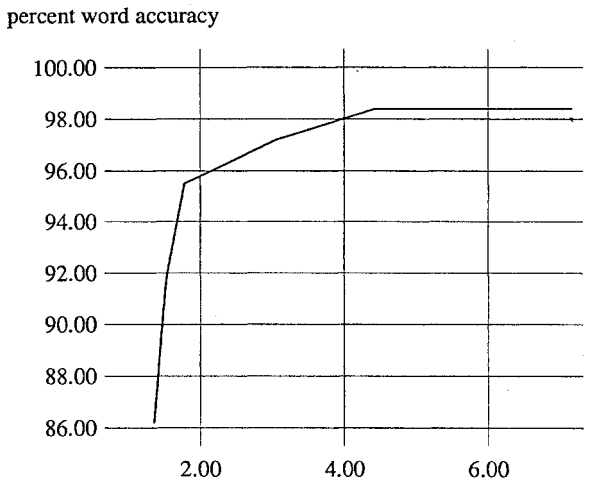

Figure 2: Word accuracy vs processing time

IMPLEMENTATION

Broca is written entirely in C and C++. This gives us maximum flexibility in developing our modules and facilitates the "integrated" approach. Everything runs in software, including the signal processing routines.

We have developed an X windows tool for browsing Broca's parse trees and other entries in its chart. The browser was used to generate figure 1. In interactive mode, the display is mouseable for audio playback of any node or segment, display of its internal data members, etc.. It is possible to get information on other entries in the parse chart and to display them in similar windows.

FUTURE WORK

We are planning to use Broca as the speech component in an interactive map system. This task will allow us to start moving in the direction of a dialog system, and to experiment with the integration of constraints involving discourse structure and world knowledge. One aim of this project will be to enhance Broca's semantic component while maintaining the principle that recognition is guided from the phoneme level on up by constraints at all levels.

Another important research area will be the incorporation of context sensitive phonological rules: for example, the rule that flaps occur only if the preceding and following phonemes are vowels. These can be easily handled in our grammatical formalism by adding mild context sensitivities that would, for example, allow the phoneme T to be expanded as the flap dx only in the appropriate contexts. We also will need to augment the phonological grammar to provide a principled account of geminate reduction, as well as other effects that cut across the constituent structure expressed in the rewrite rules.

We are also interested in exploring the incorporation of acoustic constraints at levels higher than that of the phoneme. Parameters such as peak energy, duration, and average pitch, can easily be computed for nodes in the parse chart. The ability to consider these attributes in a structured way at various levels is an advantage of integrated parsing which we have only begun to exploit. For example, rules expressing constraints on prosody and stress pattern should fit into this framework very nicely.

BIBLIOGRAPHY

[1] H. Hermansky. "Perceptual linear predictive (PLP) analysis of speech," *Journal of the Acoustic Society of America*, 87 (4) April 1990.

[2] J. R. Glass, "Finding Acoustic Regularities in Speech: Applications to Phonetic Recognition," Ph.D. thesis, M.I.T. Dept. of Electical Engineering, May 1988.

[3] K. W. Church, *Phonological Parsing in Speech Recognition*, Boston: Kluwer Academic Publishers, 1987.

[4] J. W. Creekmore, M. Fanty, R. A. Cole. "A comparative study of five spectral representations for speaker-independent phonetic recognition," *Proceedings of the 25th Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, CA, Nov., 1991.

[5] E. Barnhard, R. A. Cole. "A neural-net training program based on conjugate gradient optimization," technical report CSE 89-014, Oregon Graduate Institute of Science and Technology, Beaverton, OR. July, 1989.

[6] V. Zue, J. Glass, M. Phillips, S. Senneff. "The MIT SUMMIT speech recognition system." *Proceedings of the DARPA Speech and Natural Language Workshop*, Philadelphia, February 1989.

[7] T. Winograd. *Language as a Cognitive Process.* Addison-Wesley Pub. Co., Reading, MA. 1983.

[8] J. Earley, "An efficient context-free parsing algorithm", *Communications of the ACM.* 6:8, 1970.

[9] L. F. Lamel, R. H. Kassel, and S. Seneff, "Speech database development: design and analysis of the acoustic-phonetic corpus," *Proceedings of the DARPA Speech Recognition Workshop.* 1986.