Automatic Discovery of Contextual Factors Describing Phonological Variation

Francine R. Chen and Jeff Shrager
XEROX PALO ALTO RESEARCH CENTER
3333 Coyote Hill Road
Palo Alto, CA 94304

Abstract

In this paper we describe a method for automatically discovering subsets of contextual factors which, taken together, are useful for predicting the *realizations*, or pronunciations, of English words for continuous speech recognition. A decision tree is used for organizing contextual descriptions of phonological variation. This representation enables us to categorize different realizations according to the context in which they appear in the corpus. In addition, this organization permits us to consider simplifications such as pruning and branch clustering, leading to parsimonious descriptions that better predict allophones in these contexts. We created trees to examine the working assumption that preceding phoneme and following phoneme provide important contexts, as exemplified by the use of triphones in hidden Markov models; our results were in general accordance with the assumption. However, we found that other contexts also play a significant role in phoneme realizations.

Introduction–Context Sensitivity in Realizations

Phonologists claim that the *context* in which a phoneme occurs leads to consistent differences in how it is pronounced. For example, one phonological rule may state that the phoneme /t/ is often flapped when it is preceded and followed by a vocalic (as in "bu*tt*er"). The construction of these rules is typically an intricate process of theory formation, rule construction and then validation or disconfirmation of these rules.

In this paper we describe an approach to partially automate rule construction, allowing for a larger number of examples to be examined and checked for consistencies. Our examples come from comparing transcriptions of spoken speech with a dictionary representation of the words spoken. We shall call the dictionary pronunciation symbols *phonemes* and define the *realizations*, or *allophones*, of a phoneme to be the set of transcription symbols corresponding to that phoneme. For example, pronunciations of the phoneme /t/ include the released, flapped, and unreleased realizations as characteristically occur in "*t*ap", "bu*tt*er", and "pa*t* new", respectively. In addition, we shall refer to a context as having *values*. For example, the context stress has values primary, secondary, and unstressed.

The approach is based on automatically forming and simplifying *decision trees*. Decision trees have been used for both understanding and classification of data (Henrichon and Fu, 1969). In our application, they provide a way of using context to organize the various realizations of a phoneme. The probability of a realization varies with the context in which the phoneme occurs. Contexts which have similar realization distributions are grouped together. The decision tree thus provides a method for representing the partitions of allophones with dissimilar probabilities, based on context.

Decision trees can be formed automatically (Breiman *et al.*, 1984; Quinlan, 1986) and converted to rules (Quinlan, 1987). The problem addressed here is to construct decision trees that are appropriate for use in the construction of pronunciation rules and for predicting realizations in context. In addition, an important part of the tree induction method is the discovery of appropriate descriptive *categories* for the formation of such trees. These categories often resemble theoretical categories, such as vocalic or plosive, and define the organization of the tree.

Method

To organize the realizations of a phoneme according to context, we adapted a decision tree induction method based upon ID3 (Quinlan, 1986). The nodes of the tree represent the attribute upon which the branching is based. In a pronunciation tree, as in Figure 1, the various contexts correspond to attributes. Each branch of a tree represents a different value of an attribute. For example, in Figure 1 the context syllable boundary (SYLL-BDRY) can take on the values final (F) and initial (I) or not-initial-and-not-final (NI-NF). Associated with each leaf in the tree are the exemplars that are characterized by the context values encountered in traversing the tree from the root node to reach the leaf. The exemplars are grouped into classes, which correspond to

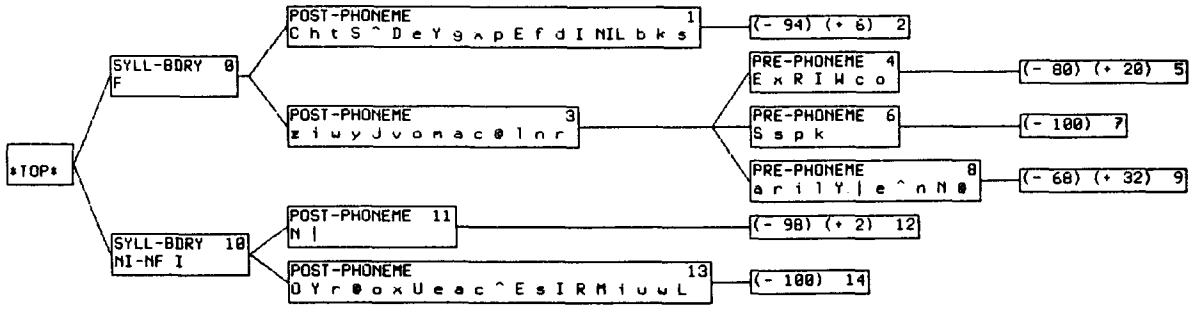

Figure 1: Pruned tree showing contexts when /t/ is glottalized (+) and not glottalized (-). Nodes are numbered in the upper right corner of each box.

realizations of a phoneme in our application, and the percentage of each realization in a leaf is paired with the class label.

Two characteristics of speech are captured by our tree representation: 1) A phoneme can be pronounced in multiple ways, as in the previous example with /t/; and 2) The number of values for some contexts may be large. For example, the context preceding phoneme has 46 values in our dictionary's alphabet. Splitting on all 46 values rapidly decreases the number of exemplars per node. These two characteristics are accommodated by using a general metric in tree induction and by using clustering of context values, both of which are described in the next sections.

Tree Induction

A decision tree is induced by recursively splitting a node into new nodes. At each node the attribute is selected which has values that best separate the realizations of the data, (*i.e.*, make each node "purer"[1]). The exemplars in the current node are subdivided according to the value of the selected attribute for each exemplar, creating a new set of nodes.

To handle multiple realizations, we used a reduction of entropy criterion with a variable number of possible classes. Intuitively, as entropy is reduced, the nodes of the tree become purer and the different realizations are better separated. We briefly review the criterion calculations here (adapted from Breiman *et al.*, 1984; Chou, 1988; Gallager, 1968; and Ganapathy and Rajaraman, 1973).

Before splitting, the entropy at a node based on classes X is $H(X)$. The average entropy of the classes in the new nodes created by splitting on the values V of a given attribute is: $E(H(X|v)) = \sum_v P(v)H(X|v)$. The average entropy can also be expressed as the conditional entropy of class X given attribute values V, or $H(X|V)$. Thus the gain for attribute a, $G(a)$, which is the difference between the entropy of the classes at a node, $H(X)$, and the conditional entropy of X given V at a node, is the mutual information between X and V: $G(a) = H(X) - H(X|V) = I(X;V)$. To normalize for the variable number of attribute values, the gain for attribute a is normalized by the entropy of the number of values associated with a. Quinlan calls this the *gain ratio*: $R(a) = G(a)/H(V)$. The attribute which maximizes the gain ratio, $R(a)$, is selected for splitting.

This decision tree induction procedure is used in our application to separate the different realizations of a phoneme based on context values. In the next section we discuss the clustering procedure which groups context values with similar realization distributions to potentially reduce the number of splits at a node and form categories.

Clustering of Attribute Values

Traditionally, in tree induction, nodes are split either along all values of an attribute (*e.g.*, Quinlan, 1986) or else binary splits are used (*e.g.*, Breiman *et al.*, 1984). When there are only a few values per attribute, splitting along all values of an attribute will not reduce quickly the number of exemplars per node. But

[1] A *pure* node contains exemplars of only one realization type.

sometimes attributes may have many different values. In speech, some of the values are thought to be similar in their influence on sound realizations; hence, in theory one would not want to split separately on all values, but instead would like to keep sounds with similar effects together. The values could be pre-clustered by a person according to theoretical ideas of what is similar, but the groupings may change depending on context. Alternatively, the values could be clustered into a predefined number of groups at each node (Chou, 1988). However, the appropriate number of groups is not the same for all sounds and again may depend on the current context. Thus, we want to cluster the values of each context at a node and want the number of clusters to be determined by the exemplars in the node. The context values within each resulting group will then be similar in their prediction of the distribution of realizations.

Hierarchical clustering is used to group the values of an attribute. This type of clustering was chosen because it allows the number of clusters for each set of attribute values to be determined from the data, rather than predefined. Mutual information is used as the distance metric and is computed as in Jelinek (1985). That is, let the average mutual information between context value or attribute value v_i and realizations or classes X be:

$$I(v_i; X) = \sum_x P(v_i, x) log_2 \frac{P(v_i|x)}{P(v_i)}$$

The increase in average mutual information resulting from pairing two attribute values v_m and v_n is the difference between the average mutual information resulting from pairing v_m and v_n and the contribution to the average mutual information before pairing v_m and v_n: $\Delta I(V; X) = I(v_m \cup v_n; X) - I(v_m; X) - I(v_n; X)$.

At each iteration, the pairing that results in the largest increase in mutual information is selected and forms a new cluster. This is continued until one of the following conditions for stopping is reached: 1) There are only two clusters left; 2) The increase in the mutual information is negative and more than doubles from one iteration to the next; 3) The increase in the mutual information measure decreases more than a threshold, which we set at -30. At each iteration, the increase in mutual information is often negative because some information is usually lost each time a new cluster is formed. The conditions for stopping define when the loss in mutual information is too great to continue clustering.

Since we cluster the values of each attribute prior to splitting, it may be useful to split on this attribute again under a more specific context (*i.e.*, farther down the tree). Thus, in contrast to ID3, after an attribute is selected for splitting, it is *not* removed from the set of attributes considered. This is also the case in the binary split method of Breiman *et al.* (1984); however, our method has the potential of providing meaningful splits which accommodate graded categorization in the realizations.

Pruning of Trees

A tree that has been constructed by this method may be too specialized to the training exemplars. In the extreme case, each leaf is pure, which is not desirable because such a tree would not be robust with respect to new data. In understanding the relationship between contexts and realizations, we want to uncover generalizations. This can be achieved by pruning, which combines subtrees, resulting in more general distinctions. Many methods of pruning have been suggested (*e.g.*, Breiman *et al.*, 1984 and Chou, 1988), but in general their primary goal is to optimize the probability of error versus some characteristic of the tree, such as average length or number of leaves.

Since our concern is to use only the parts of the tree which will be robust to new data, a different type of pruning was used. First, nodes are extended only when the number of exemplars is greater than a specified threshold (Breiman *et al.*, 1984); we used 20. In addition, only nodes relevant to the classification of exemplars are kept. A chi-square test (Quinlan, 1986) at the .01 level of significance is used. Each tree is also pruned by running a separate set of exemplars, or cross-validation set, through the constructed tree. If the cross-validation exemplars in a node indicate that an attribute is not relevant to the classification of the exemplars in a node, the subtree beginning at the node is collapsed into a leaf.

Data

The data comprised almost 30,000 hand-transcribed segments from approximately 900 of the "sx" sentences from the TIMIT acoustic-phonetic speech database (Lamel *et al.*, 1986; Fisher *et al.*, 1987), spoken by more than 180 different speakers. Trees were induced using 60% of the data; the results that will be described are based on trees pruned on the remaining 40% of the data. The transcribed TIMIT data were automatically

context	values
preceding phoneme	(all phonemes)
following phoneme	(all phonemes)
syllable part	onset, nucleus, coda
stress	primary, secondary, unstressed
syllable boundary type	initial, final, not-initial-and-not-final, initial-and-final
foot boundary type	initial, final, not-initial-and-not-final, initial-and-final
word boundary type	initial, final, not initial-and-not-final, initial-and-final
cluster type	onset, coda, nil
open syllable?	true, false
true vowel?	true, false
function word?	true, false

Table 1: Contexts used in pronunciation experiments

aligned to the dictionary baseforms from the *Merriam-Webster 20,000 Word Pocket Dictionary* to produce *mappings* between dictionary phonemes and transcribed segments. Each mapping was then described by a set of theoretically motivated contexts based on a set used by Withgott and Bagley (1987) in a pronunciation generation system. These contexts and corresponding values, which are the union of possible values over all phonemes, are listed in Table 1. Note that some contexts, such as stress and foot-boundary refer to how the phoneme functions within a larger unit. Because lexical context is used, contexts such as foot-boundary are easily determined. Also note that the context values based on adjacent phonemes are defined across word boundaries. For example, in the phrase "two words" the post (following) phoneme to /u/ would be /w/. Some of the common predicate rule contexts were combined into a context *type* and the clustering algorithm was used to group these values to form predicates when appropriate. For example, the predicates syllable-initial?, syllable-final?, and syllable-internal? were combined into the type syllable-boundary. The predicate syllable-initial? is formed when the values of syllable-boundary-type are clustered into the groups {initial} and {final, not-initial-and-not-final, initial-and-final}.

Results

A sample tree constructed using the previously described method is shown in Figure 1. This tree is for the special case describing whether /t/ is glottalized. A context other than preceding phoneme and following phoneme is incorporated; the first split (nodes 0 and 10) in this tree is on syllable boundary (SYLL-BDRY), indicating that when /t/ is glottalized it is generally in syllable-final position. Note that grouping the syllable boundary context values not-initial-and-not-final (NF-NI) and initial (I) separately from final (F) can be interpreted as the predicate syllable-final?. Without clustering, selection of this attribute would result in a three-way split. Further examination of this tree suggests additional conditions on this generalization. For example, node 3 may be described as containing a subset of voiced sounds; in particular all semivowels and nasals and most of the voiced fricatives and tense vowels. Node 1 contains primarily plosives, unvoiced fricatives, and lax vowels. Additionally, the nodes labeled 5 and 9 are preceded by PRE-PHONEME nodes containing vocalics. Thus, we might produce a more precise rule, predicting that a /t/ in syllable-final position and preceded by a vocalic will be glottalized with greater likelihood when preceded by a phoneme in node 8 than in node 4.

Along with trees such as these, we have constructed 45 general trees, one for each phoneme in the dictionary. These trees were more general in that all the realizations of a phoneme composed the classes. Twenty of the trees (p t k b d g m n w h u U i | I E x c s T) had preceding phoneme and following phoneme as their initial contexts for splitting; in nine trees (C J r l y e ʌ Z D) preceding phoneme but not following phoneme appeared in the first two levels; and in eight trees (G o Y @ a z f v) the following phoneme but not preceding phoneme appeared in the first two levels. This agrees with the common working assumption that preceding phoneme and following phoneme are the most important contexts for describing phonological variation. However, we also observed that other contexts are useful for differentiating among the realization distributions. The additional contextual factors which appeared in the first two levels of the tree and the

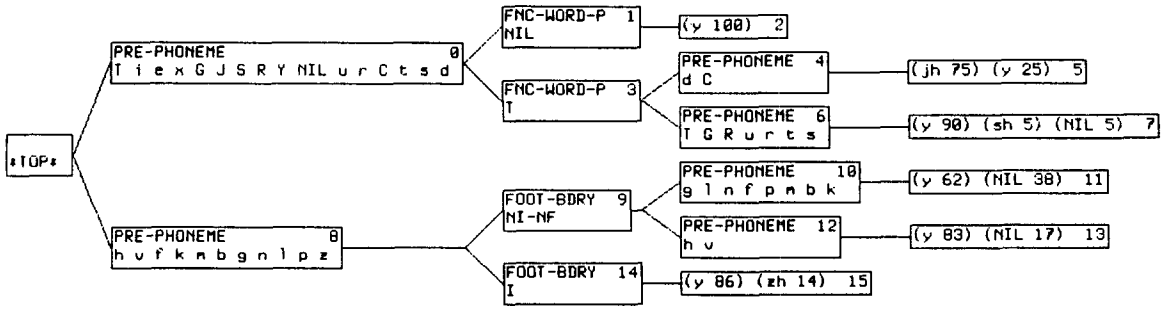

Figure 2: Pruned tree of /y/ realizations. Nodes are numbered in the upper right corner of each box.

number of times each appeared are: stress 13, function-word? 8, foot boundary type 5, syllable boundary type 5, syllable part 3, open syllable? 1, and word boundary type 1.

By using a clustering technique in which the number of groups was determined by the data, many times the preceding and following context categories corresponded to linguistic categories, such as place or manner. For example, the POST-PHONEME values in the /s/ realizations were clustered into the set $\{\check{s}, \check{c}, y\}$ and the set of all other phonemes. In addition, the predominance of preceding phoneme and following phoneme as useful contextual factors is due in part to the flexibility in the number of groups.

Discussion–Relevance to Speech Recognition

Contexts used in the creation of pronunciation networks for some hidden Markov model speech recognition systems have been limited. This is partially due to the amount of data needed to train the network units in which context is represented. These units include whole word models, where phones are represented in the context of the word in which they occur (*e.g.*, Paul, 1988), generalized triphones (Lee, 1988), and a hierarchy from words to subsets of triphones (Chow *et al.*, 1986). Whole word models provide the most complete context of the internal phones, but usually do not model word boundary effects well. Although a subword unit, such as the triphone, can be concatenated into word models, thus providing easy additions to the lexicon, triphones account for only a subset of contextual factors. Work using log-linear modeling (Chen, 1987) has shown that use of only the preceding and following contexts do not adequately describe realization distributions.

To test the common working assumption that preceding phoneme and following phoneme are the most useful contextual factors, as in triphone models, we examined the contexts in the trees constructed for each phoneme. As stated in the previous section, when the realizations of a phoneme are considered jointly, the preceding phoneme and following phoneme are the most useful contexts *overall.* However, additional contextual factors other than preceding phoneme and following phoneme can provide better estimates of the likelihood of different realizations. We thus suggest a mixed context unit based on the partitioning represented by the computed trees for use in continuous speech recognition. The organization of the phonological realizations into trees provides a way to specify contexts for creating models intermediate in the continuum of context models from adjacent phone to whole word. A subset of a predetermined set of possible contexts which are useful for differentiating among the realization distributions is identified. This subset is a larger number of contexts than the data would permit if the selected contexts were always considered together. Consequently, a larger overall number of contexts can be used for describing the realizations. For example, in Figure 2, the contexts of PRE-PHONEME, FNC-WORD-P, and FOOT-BDRY are used for describing the realizations of /y/, but only two contexts, either PRE-PHONEME and FNC-WORD-P or PRE-PHONEME and FOOT-BDRY, are used to describe each leaf.

In tree induction, different realizations are considered simultaneously and the partitioning based on context values is mutually exclusive. By considering all realizations of a phoneme simultaneously, the *overall* usefulness of the different contextual factors is analyzed. The set of selected contexts generally is not the same as those chosen in trees in which the occurrence of each realization of a phoneme is separately computed. Since each exemplar belongs to one set of context values because of the partitioning, the proportion of each realization for each set of context values can be estimated.

Conclusions

In our work describing phonological variation for speech recognition, we use a systematic, data-intensive approach. Contexts are identified that correlate with the phonological variation exhibited in a large hand-transcribed database of utterances. From these correlations, we identify useful context descriptions. The combination of decision tree induction and hierarchical clustering organizes the realization data into a representation conditioned on context. The tree induction attempts to separate different realizations, while the hierarchical clustering provides for theoretically meaningful grouping of the context values, which, in turn, allows for better estimates of the realization distributions. The use of a tree structure allows multiple mutually exclusive context sets to be used to describe allophone distributions. The trees can be traversed to produce pronunciation distributions for each phoneme in a dictionary baseform. Because the chosen contextual factors in a tree are dependent on contextual factors chosen closer to the root node (*e.g.*, for the phoneme /y/ in Figure 2, FOOT-BDRY is useful when PRE-PHONEME has the values {h, v, f, k, m, b, g, n, l, p, z}), the context trees, rather than the set of contexts in the trees, should be used for building pronunciation networks for speech recognition systems. The context trees can be reinterpreted straightforwardly, and we are examining the clustered context values in each branch for such general descriptions.

Acknowledgement

This work was sponsored in part by the Defense Advanced Research Projects Agency (DOD), under the Information Science and Technology Office, contract #N00140-86-C-8996.

References

L. Breiman, J.H. Friedman, R.A. Olshen, and C.J. Stone, *Classification and Regression Trees*, Wadsworth International Group, Belmont, CA, 1984.

F. Chen, "The importance of context on the realization of phonemes," *J. Acoust. Soc. Am.*, Suppl. 1, vol. 82, 1987.

P. Chou, *Applications of Information Theory to Pattern Recognition and the Design of Decision Trees and Trellises*, Doctoral Dissertation, Stanford University, Stanford, CA, June 1988.

Y.-L. Chow, R. Schwartz, S. Roucos, O. Kimball, P. Price, F. Kubala, M. Dunham, M. Krasner, J. Makhoul, "The role of word-dependent coarticulatory effects in a phoneme-based speech recognition system," *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Proc.*, pp. 1593-1596, 1986.

W. Fisher, V. Zue, J. Bernstein, D. Pallett, "An acoustic-phonetic data base," *J. Acoust. Soc. Am.*, Suppl. 1, vol. 81, 1987.

R. Gallager, *Information Theory and Reliable Communication*, John Wiley and Sons, New York, 1968.

S. Ganapathy and V. Rajaraman, "Information theory applied to the conversion of decision tables to computer programs," *Commun. of the ACM*, vol. 16, no. 9, pp. 532-539, 1973.

J. Henrichon and K. Fu, "A nonparametric partitioning procedure for pattern classification," *IEEE Transactions on Computers*, vol. C-18, pp. 604-624, May 1969.

F. Jelinek, "Self-organized language modeling for speech recognition," unpublished, IBM T.J. Watson Research Center, Yorktown Heights, N.Y., 1985.

L. Lamel, R. Kassel, S. Seneff, "Speech database development: design and analysis of the acoustic-phonetic corpus," *Proceedings of the DARPA Speech Recognition Workshop*, L. Baumann, ed., pp. 100-109, 1986.

K.-F. Lee, *Large-Vocabulary Speaker-Independent Continuous Speech Recognition: The SPHINX System*, Doctoral Dissertation, Carnegie Mellon University, Pittsburgh, PA, April 1988.

D. Paul, "Speaker stress-resistant continuous speech recognition," *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Proc.*, pp. 283-286, 1988.

J.R. Quinlan, "Induction of decision trees," *Machine Learning*, Kluwer Academic Publishers, Boston, vol. 1, pp. 1-86, 1986.

J.R. Quinlan, "Generating production rules from decision trees," IJCAI-87, pp. 304-307, 1987.

M. Withgott and S. Bagley, The Variant Pronunciation Rule System, (implementation), Xerox Palo Alto Research Center, Palo Alto, CA, 1987.