Hierarchical Phoneme Recognition Using Node-wise Relevance-Optimized Features

Ali E. Hameed
e-mail: ali_essam_hameed@yahoo.com

Intessar T. Hwaidy
e-mail: intessar_hwaidy@yahoo.com

Department of Computer Engineering, College of Engineering, University of Basrah

ABSTRACT

In this paper, a hierarchical phoneme recognition system is proposed. The hierarchical approach is applied here to recursively partition the recognition problem into smaller and smaller sub-problems those are independently handled at the distinct nodes of the hierarchy. The nodes are individually set to characterize different properties of the input phoneme, or more precisely to make separate decisions on its pertinence to the different reference subgroups of phonemes. The full characterization of the input phoneme is achieved by traversing some root-to-leaf path through the hierarchy. The relationships between the different features of phonemes and their pertinence to the different reference subgroups are to be objectively characterized and optimized here. This involves specifying the decisive subset of features for each pertinence decision and neglecting the remaining features those are irrelevant to (or probably have negative effect on) that decision, at each node of the hierarchy. The optimization applied through the feature election process here, is not aimed at reducing the amount of features to be used in the recognition process, for the purpose of decreasing the time-complexity of the system, but, is interested in enhancing the decision making accuracy of the system by avoiding the misleading features.

KEYWORDS: Phoneme recognition, hierarchical systems, neural networks, Mel-cepstrum, information relevance.

التمييز المهيكل للأصوات باستخدام الخواص المثلى من حيث الصلة عند كل عقدة هيكلية

علي عصام حميد

إنتصار طعيس هويدي

قسم هندسة الحاسبات، كلية الهندسة، جامعة البصرة

الخلاصة

في هذا البحث، تم اقتراح نظام مهيكل لتمييز الأصوات، يقوم على أساس تقسيم عملية التمييز، بشكل متكرر، إلى مجموعة من القرارات الأصغر و الأسهل والتي يمكن معالجتها بشكل مستقل عند العقد المختلفة في التركيب الهيكلي للنظام. بهذا التقسيم يصبح من الممكن أن تُخصص كل عقدة بتمييز خواص مختلفة من الصوت المُدخل، أي باتخاذ قرار منفصل حول انتماء هذا الصوت إلى أحد التصنيفات الجزئية للأصوات. أما التمييز الكلي للصوت المُدخل فيتم من خلال المرور بمسار يمتد من جذر الهيكل إلى أحد أغصانه الطرفية. لقد تم، في هذا البحث، إيجاد الخواص المثلى المميزة لعناصر كل صنف من الأصوات على حدة، من أجل التقليل من التأثير السلبي للخواص الأخرى غير ذات الصلة (أو المناقضة) على قرارات التمييز الجزئية. إن تحديد عدد المميزات المستخدمة في عملية التمييز هنا لا تهدف إلى زيادة سرعة أداء النظام، و إنما إلى زيادة دقته في تمييز الأصوات.

I. Introduction

The basic idea involved in any multistage hierarchical decision making approach is to break up a complex decision into a union of several simpler decisions [1], each of which is to be tackled independently. This approach is called *divide and conquer* [2].

The hierarchical phoneme recognition system, proposed in this paper, partitions the phoneme recognition process into a hierarchy of partial recognition decisions those are carried out at the distinct *nodes* of the hierarchy. A partial decision determines the probability of pertinence of the incoming phoneme to each reference subgroup of phonemes branching from the corresponding node. By further dividing each partial decision into sub-decisions, each of which being involved in a smaller subgroup of phonemes, a full conclusion about the incoming phoneme is reached by traversing some root-to-leaf path (or paths) through the hierarchy.

Hierarchical decision making provides the flexibility to choose different deciding rules at the different nodes of the hierarchy. In the context of phoneme recognition, this means the capability to use different subsets of features, of the input phoneme, at the different decision making nodes of the hierarchy [1, 3, 4]. The proposed hierarchical recognition system takes advantage of this capability to separately optimize the subset of features to be used in making the partial decision at each node of the hierarchy.

The optimization process to be applied here includes objectively identifying the features those contribute to increasing both the *discrimination* among the distinct subsets of phonemes, to be decided upon at each node, and the *similarity* among the individual phonemes within each subset, alone. Adapting this feature *pruning* process to the distinct parts of the problem (corresponding to the different nodes of the hierarchy) should increase the accuracy of targeting the unwanted features without having to compromise those ones that show high relevance at some parts of the problem, but also misleading behavior at other parts of it. This will minimize the overall effect of the misleading features at the different levels of building a conclusion.

II. The Proposed Hierarchical Recognition Tree

In designing hierarchical tree recognizers, it is important to adapt an appropriate tree structure that reflects the natural divisions among the reference objects. The simplest way to map these divisions onto a tree structure is to apply a *hard-split* on the set of reference objects, such that the subsets resulting from each division have no common elements among them [3, 4].

The tree structure to be adopted in this work is primarily based on prior knowledge with *Arabic phoneme* classes and common sense similarities and dissimilarities among these phonemes [5]. No objective optimization of any kind has been applied to conclude this tree. It is expected, though, that such optimizations could improve the overall performance of the system, and probably the effectiveness of the feature pruning process, too. The adopted tree structure is shown in Fig. 1, using International Phonetic Association (IPA) notation.

As is shown in Fig. 1, two types of decisions are encountered throughout the proposed tree, namely: binary decisions (about the pertinence of the incoming object to one of two reference subgroups), and non-binary decisions (about the pertinence of the incoming object to one of more than two reference subgroups). Binary decisions are handled, each, by a single decision maker with a binary result space. Non-binary decisions, on the other hand, are handled, each, by a union of binary decision makers, each of which determines whether the incoming object is pertinent or not to one of the descending reference subgroups (one decision maker per one class [6]).

In this work, the decision making process, performed at the distinct nodes of the tree, is proposed to be handled by *multi-layer perceptron (MLP) artificial neural networks*

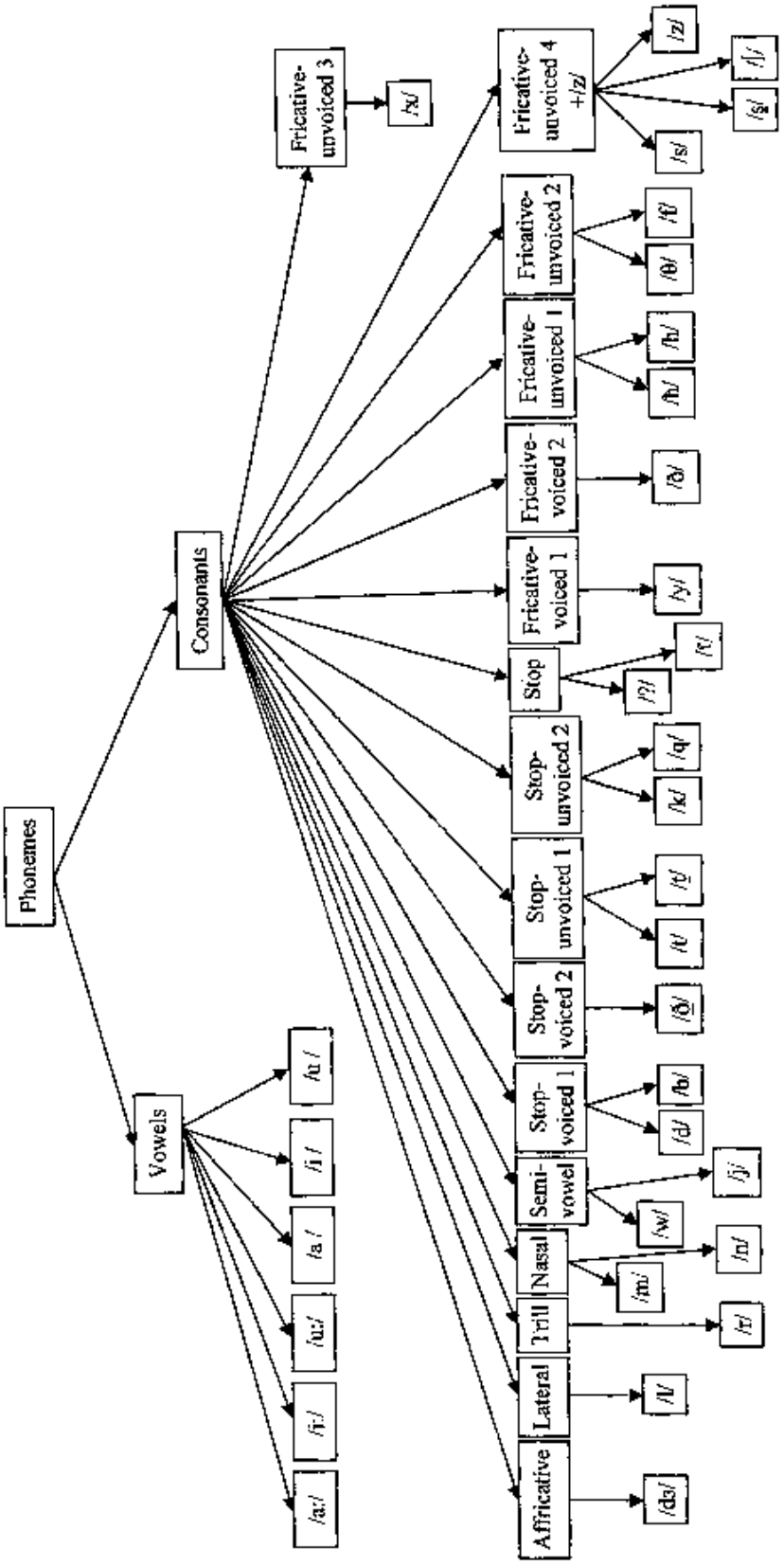

Fig. 1: The proposed phoneme recognition tree.

with *error backpropagation*. The networks are to be trained using the *resilient backpropagation algorithm* [7, 8]. This algorithm is well-known for being a fast learning algorithm, and has been achieving good results in the field of pattern recognition [9].

III. Phoneme Database

In this paper, we will consider 33 different Arabic phonemes (in many Arabic accents, the letter corresponding to the /d̲/ phoneme is pronounced as /ð/), 27 of which are consonants, and the remaining 6 are vowels (including 3 long and 3 short vowels). Most of the contextual versions of phonemes in Arabic language come in syllables involving a consonant followed by or following some short or long vowel [10].

In order to address the co-articulation effects among the individual phonemes in Arabic language syllables, 12 different data tokens per phoneme are included in our training database. These tokens correspond to different vowel-consonant and consonant-vowel permutations [10]. The training database involves a total of 33 * 12 = 396 tokens, in the form of pulse code modulated (PCM) wave (.wav) sound files. An entirely separate database of other 396 tokens is used for testing.

The voice recording process has been carried out in a relatively quiet room, using regular non-expensive recording equipment. Also, no special noise elimination or reduction processing has been applied on the recorded voice files.

IV. Preprocessing

Preprocessing includes all data processing being applied to the recorded voice files in a database, in order to prepare them for the feature extraction phase.

The first step in the preprocessing phase is the segmentation process. Here, the precise phoneme parts are cropped out of the recorded signals of the corresponding phoneme tokens, and are saved for further processing. The segmentation in this work is carried out manually in order to overcome automation inaccuracy.

The next preprocessing step is filtering. This process is mainly aimed at reducing the effect of the noise induced by the recording process. A bandpass IIR elliptic filter of the 10^{th} degree with a passband of 160 Hz – 6.8 kHz is used to handle this process here (the characteristics of this filter has been concluded empirically).

The last preprocessing step is framing. Here, each segmented and filtered voice token is divided into non-overlapping frames of 128 data samples (corresponding to 8 msec duration, given the sampling rate, 16 kHz, of our recording process). The number of frames extracted out of each token depends on its duration.

V. Feature Extraction

Finding an efficient data representation that reflects the natural characteristics of phonemes has been a major concern in the field of phoneme recognition. Many different feature classes have been proposed to be used. In this work, we will use *Mel-cepstrum* (also known as *Mel-frequency-cepstrum coefficients MFCC*) features of phoneme tokens.

The Mel-cepstrum transformation, here, is applied to each token in a frame-wise fashion (that is to each frame of the given token independently), in order to extract an N-dimensional feature vector (corresponding to the N transformation coefficients) out of each token frame. The feature vectors extracted from the frames of a given token are then averaged into one N-dimensional vector, with the individual vectors being given hamming shaped weights, in order to give more importance to the mid frames of the token, without entirely neglecting the information from the boundaries. The resulting feature vector is then normalized in order to remove loudness information. Deferring the normalization process to this point is meant to assure that the feature vector has unit energy, regardless of the features constituting that vector (different features will be used at each node

of the recognition tree, as will be seen in the next section).

VI. Node-wise Optimization Process

In order to optimize the features used in any decision making process, it is necessary to be able to separate the features containing useful information from those containing contradicting or irrelevant information, with respect to the decision to be made. This is especially true in the case of recognition decisions, where only a part of the information contained in the generated features for the considered set of reference objects is responsible for determining the reference class that each object belongs to, whereas the remaining information is either contradicting, in the sense that it points out to the similarities among objects belonging to different classes, or irrelevant, in the sense that it says nothing about the specified division among the reference objects under consideration. Unfortunately, it is not possible, in practice, to accurately apply such a precise classification on any objectively-generated features, since that they usually involve a fused blend of the aforementioned classes of information.

The feature pruning scheme, proposed in this work, approximates the relevance of features by their varying patterns over the set of objects under consideration, given some predefined division on this set. The features dominated by useful information are assumed, here, to be those showing a higher average discrimination among the objects belonging to different classes than among those within the same classes. On the other hand, the features dominated by contradicting information are assumed to be those behaving the exact opposite way, by showing a higher average discrimination among the objects within the same classes than among those belonging to different classes. The features dominated by irrelevant information or equally dominated by both useful and contradicting information would, however, show the same level of average discrimination among the objects belonging to the same or to different classes. That is, given a binary decision about the pertinence to one of two sets of objects

$$A = \{\mathbf{a}_1^N, \mathbf{a}_2^N, \ldots, \mathbf{a}_K^N\}, \text{and}$$

$$B = \{\mathbf{b}_1^N, \mathbf{b}_2^N, \ldots, \mathbf{b}_L^N\},$$

where each object is an N-dimensional feature vector (corresponding to the N transformation coefficients), then the overall relevance R_i of the ith feature $\mathbf{x}_j^N(i)$ (the ith dimension of each feature vector $\mathbf{x}_j^N$, where $\mathbf{x}$ is either $\mathbf{a}$ or $\mathbf{b}$, and j runs through all the elements of A and B) is the average discrimination between any pair of elements belonging to two different groups Vd_i, minus the average discrimination between any pair of elements belonging to the same group Vs_i. That is

$$R_i = Vd_i - Vs_i, \quad \text{for } i = 1, 2, \ldots N,$$

with

$$Vd_i = \frac{\sum_{p=1}^{K}\sum_{q=1}^{L} d(\mathbf{a}_p^N(i), \mathbf{b}_q^N(i))}{KL}, \text{ and}$$

$$Vs_i = \frac{\sum_{p=1}^{K-1}\sum_{q=p+1}^{K} d(\mathbf{a}_p^N(i), \mathbf{a}_q^N(i))}{\binom{K}{2}+\binom{L}{2}} + \ldots$$

$$\ldots + \frac{\sum_{p=1}^{L-1}\sum_{q=p+1}^{L} d(\mathbf{b}_p^N(i), \mathbf{b}_q^N(i))}{\binom{K}{2}+\binom{L}{2}},$$

where the discrimination function $d(x, y)$ is defined as

$$d(x,y) = 2\cdot|x-y|/(|x|+|y|),$$

and $\binom{K}{2}$, $\binom{L}{2}$ are the numbers of combinations of K and L elements, respectively, taken two. A positive value of R_i indicates that the corresponding feature is dominated by useful information, a negative value, on the other hand, indicates that it is dominated by contradicting

information, while a zero, indicates that the corresponding feature is irrelevant. The tradeoff between information content and time-complexity may be controlled through varying the pruning threshold around zero. In this work all the features with positive overall relevance R_i are considered to contain useful information.

Note that the proposed feature pruning scheme does not decompose the information content of the features, but rather, it tries to classify them according to their overall dominant behavior. The performance of this scheme heavily depends on the ability of the used data transformation to decompose the information content of the transformed data, given some partial recognition decision. This makes it necessary to use the best possible data transformation for generating the features needed to make each decision.

The performance F of a data transformation, with respect to some partial recognition decision, can be determined by averaging the overall relevance values for the useful features (only the features with positive R_i, here). That is

$$F = \underset{i}{Av}(R_i), \text{for all } i \text{ where } R_i > 0$$

In this work, different resolutions of the Mel-cepstrum transformation are used for generating a number of the feature vectors databases for each node of the recognition tree. The resolution with the optimal performance index, for each node, is then selected, and the corresponding database is then used to train that node. This node-wise transformation optimization process needs to be performed only once, prior to training the recognition system, and hence, has no effect on its time-complexity. The complexity of the feature extraction process, though, can still be increased by the use of different transformations at the different nodes. However, it should always be remembered that fewer features need to be generated, due to the pruning process, and that only a part of the nodes within the tree needs to be passed through.

VII. Simulation Results

In order to recognize the phoneme content of a preprocessed voice frames set, the recognition system needs to pass this set into the nodes of the recognition tree, starting from the root node (the consonant-vowel node in Fig. 1), traversing a decision-controlled path through the tree, till a leaf is reached. Each time a decision node is reached, the corresponding transformation is applied to the voice frames set in order to generate the required features for making that decision. Only the useful features need to be generated at each node. The features are then passed to the inputs of the decision making neural network (or networks in case of non-binary decisions) in order to determine the subgroup that the phoneme belongs to (the path to be taken to the successor node in the tree). Only hard-decision scheme is tested in this work. The proposed system, however, can easily support the soft-decision scheme.

The recognition accuracies for the 33 distinct Arabic phonemes and the overall recognition accuracy of the proposed hierarchical recognition system with node-wise relevance-optimized features are shown in Table 1.

VIII. Conclusions

In this paper, we have proposed to apply the divide and conquer approach to break up the problem of phoneme recognition into a tree-like hierarchy of partial recognition decisions, in order to provide the ability to optimize the features used in decision making, independently, at each node of the recognition tree. This feature optimization process involves both choosing the best possible data transformation, and eliminating any unuseful features. The discrimination that the features show among the phonemes belonging to different and to the same classes is used, here, to determine the features relevance to the corresponding partial recognition decisions.

Table 1: The recognition accuracy of the proposed system compared to other systems.

Phoneme	[illegible]	The Hierarchical sys. [illegible]	Hierarchical recognition [illegible] proposed in [illegible]	Flat recognition [illegible] proposed in [illegible]
ء	/ʔ/	50	75	50
ب	/b/	83.33	58.33	66.67
ت	/t/	75	75	50
ث	/θ/	83.33	58.33	50
ج	/dʒ/	91.67	75	50
ح	/ħ/	75	83.33	83.33
خ	/x/	50	66.67	66.67
د	/d/	66.67	58.33	33.33
ذ	/ð/	58.33	66.67	33.33
ر	/r/	83.33	66.67	66.67
ز	/z/	91.67	100	75
س	/s/	91.67	50	50
ش	/ʃ/	100	100	75
ص	/s̲/	83.33	66.67	50
ظ	/ð̲/	75	16.67	8.33
ط	/t̲/	41.67	66.67	41.67
ع	/ʕ/	83.33	75	50
غ	/γ/	58.33	50	25
ف	/f/	75	66.67	75
ق	/q/	25	50	25
ك	/k/	58.33	58.33	33.33
ل	/l/	83.33	75	58.33
م	/m/	100	66.67	50
ن	/n/	75	33.33	41.67
هـ	/h/	91.67	58.33	33.33
و	/w/	100	83.33	50
ي	/j/	83.33	100	50
آ (علة)	/a:/	100	100	75
ي (علة)	/i:/	100	91.67	75
و (علة)	/u:/	100	41.67	33.33
(فتحة)	/a/	100	83.33	75
(كسرة)	/i/	58.33	58.33	41.67
(ضمة)	/u/	66.67	75	50
Total Accuracy		[illegible]	68.18	51.[illegible]

The proposed hierarchical recognition system with node-wise relevance-optimized features has achieved an overall recognition accuracy of 77.53 %, a significant improvement in terms of recognition accuracy, when compared to the previous hierarchical and non-hierarchical (flat) recognition systems, such as the systems proposed in [3, 4], which have achieved overall recognition accuracies of 68.18 % and 51.26 %, respectively, as is shown in Table 1.

The proposed system has also managed to overcome the difficulties suffered by the previous two systems concerning the recognition of some of the phonemes, such as the / ð /, / u: /, and / n / phonemes.

REFERENCES

[1] S. R. Safavian and D. Landgrebe, "A Survey of Decision Tree Classifier Methodology", *IEEE Transactions on Systems, Man, & Cybernetics*, vol. 21, no. 3, pp. 660-674, May 1991.

[2] S. R. Waterhouse, "Divide and Conquer Pattern Recognition using Mixtures of Experts", Ph.D. Dissertation, *Jesus College, University of Cambridge, and Cambridge University Engineering Department*, Feb 1997.

[3] A. A. Ali and I. T. Hwaidy, "Hierarchical Arabic Phoneme Recognition using MFCC Analysis", *Iraqi Journal for Electrical and Electronic Engineering*, vol. 3, no. 1, pp. 97-106, 2007.

[4] I. T. Hwaidy, "Hierarchical Arabic Phoneme Recognition using Neural Networks", M.Sc. Thesis, *Computer Engineering Department, Engineering College, University of Basrah*, March 2006.

[5] L. Rabinar and R. W. Schafer, "Fundamentals of Speech Recognition", *Prentice Hall*, 1993.

[6] J. Y. Siva Rama Krishna Rao, "Recognition of Consonant–Vowel (CV) Utterances using Modular Neural Network Models", M.Sc. Thesis, *Department of Computer Science and Engineering, Indian institute of Technology, Madras*, May 2000.

[7] M. Riedmiller and H. Braun, "A Direct Adaptive Method for Faster Backpropagation Learning: The RPROP Algorithm", *Proceedings of the IEEE International Conference on Neural Network, San Francisco*, pp. 586-591, 1993.

[8] M. Riedmiller, "Advanced Supervised Learning in Multi-layer Perceptrons – from Backpropagation to Adaptive Learning Algorithms", *International Journal of Computer Standard and Interface*, vol. 16, pp. 265-278, 1994.

[9] P. Melin, J. Urias, D. Solano, M. Soto, M. Lopez and O. Castillo, "Voice Recognition with Neural Networks, Type-2 Fuzzy Logic and Genetic Algorithms", *Engineering Letters*, 13:2, *EL_13_2_9*, *Advance online publication*, August 2006.

[10] A. A. Jasim, "Hybrid Approaches for Arabic Phoneme Recognition", M.Sc. Thesis, *Computer Engineering Department, Engineering College, University of Basrah*, 2004.