HIERARCHICAL ARABIC PHONEME RECOGNITION USING MFCC ANALYSIS

PROF. DR. ABDULADHEM A. ALI   INTESSAR T. HWAIDY
e-mail:Abduladem1@yahoo.com   e-mail:intessar_hwaidy@yahoo.com
Department of Computer Engineering, Engineering College,
University of Basrah, Basrah, Iraq

ABSTRACT

In this paper, a hierarchical Arabic phoneme recognition system is proposed in which Mel Frequency Cepstrum Coefficients (MFCC) features is used to train the hierarchical neural networks architecture. Here, separate neural networks (subnetworks) are to be recursively trained to recognize subsets of phonemes. The overall recognition process is a combination of the outputs of these subnetworks. Experiments that explore the performance of the proposed hierarchical system in comparison to non-hierarchical (flat) baseline systems are also presented in this paper.

Keywords: speech recognition, phoneme recognition, hierarchical systems, Mel cepstrum, neural networks

التمييز المهيكل للاصوات العربية بأستخدام الشبكات العصبية

الاستاذ الدكتور عبد العظيم عبد الكريم علي   انتصار طعيس هويدي
قسم هندسة الحاسبات - كلية الهندسة - جامعة البصرة

الخلاصة

في هذا البحث تم اقتراح نظام مهيكل (hierarchical) لتمييز الاصوات العربية حيث تم استخدام معاملات (Mel Frequency Cepstrum Coefficients) لتدريب مجموعة من الشبكات العصبية المهيكلة من خلال تدريب كل شبكة عصبية مفردة (subnetwork) على حدة وبصورة تكرارية لجعلها قادرة على تمييز مجاميع محددة من الاصوات العربية. النتيجة النهائية لعملية التمييز يمكن استنتاجها من خلال اخراجات هذه الشبكات العصبية. وقد اظهرت التجارب التي اجريت لاختبار اداء النظام المهيكل المقترح بالمقارنة مع النظام غير المهيكل (flat) ان استعمال التركيب المهيكل يزيد دقة تمييز الاصوات العربية.

1. INTRODUCTION

Phoneme recognition can be defined as the process where an input phoneme signal is assigned to one of the prescribed phoneme classes. The training of a monolithic neural network classifier that performs the class recognition based on All-Class-One-Network (ACON) architecture is practically difficult especially when the number of classes is large and the similarity among them is high. Classifiers have been developed with the One-Class-One-Network (OCON) architecture, in which a separate network is trained for each class [1]. However, the discrimination capabilities of the OCON classifiers are still poor [2, 3].

In this paper a hierarchical approach is proposed to overcome the limitations of ACON and OCON architectures. Under hierarchical approaches, the large number of classes is grouped into fewer subgroups with a separate neural network being trained for each subgroup [2].

The proposed hierarchical recognition system (built as a collection of neural networks) forms a tree like structure, in which many paths can be traversed from the root node down to the terminal nodes (leaves). It is based on the principle of "Divide and Conquer", where a large problem is recursively divided into smaller and easier problems, whose solutions can be combined to yield a solution to the overall complex problem [4, 5]. In contrast to conventional non-hierarchical (flat) baseline recognizers, where each data sample is tested against all possible classes, which could sufficiently reduce the recognition efficiency, in a hierarchical tree recognizer a sample is tested against only certain subsets of classes, thus, eliminating unnecessary computations. Hierarchical Tree Recognizers (HTRs) have the flexibility of choosing different feature subsets and making decision rules at the different stages of recognition, in addition to the capability of trading-off between recognition accuracy and time-space efficiency. However, in large tree recognizers, the effect of recognition errors may grow from level to level, which can be considered a significant drawback to HTR that points to the fact that one cannot simultaneously optimize both the accuracy and the efficiency of a system; since that for any given accuracy, a bound on the efficiency must be satisfied. Moreover, difficulties might be encountered in designing an optimal HTR. The performance of a HTR strongly depends on how well the tree is designed [6].

2. Design of the Hierarchical Phoneme Recognition System

In designing HTR, it is important to search for the appropriate tree structure and the feature subsets to be used at each node. Various methods for subdividing large problems have been proposed in the literature. The simplest approach is to divide the problem into sub-problems that have no common elements, also called a “hard split” [6]. The proposed tree here has partly been based on the prior knowledge of phoneme classes. This is motivated by the observation that these classes are easy to be distinguished acoustically. At the root of the tree, consonant and vowel classes are to be discriminated. Both consonant and vowel classes are further split into subsets of classes. Vowels are split into six leaves (terminal nodes) that represent long and short vowels in Arabic language. Consonants can be further divided into voiced and unvoiced groups according to the manner of articulation; hence, branching can be continued until each branch in the tree is assigned only one phoneme

2.1 Phonemes Database

Arabic language involves 33 different phonemes, 27 phonemes of them are consonants, and the remaining 6 are vowels. The phonemes / d̲ / and / ð̲ / are considered as one phoneme in this work because they have similar pronunciation in local Arabic language. Most of the contextual versions of the consonants in Arabic language can be obtained by considering the syllables in which consonants are followed by (or are following) some short or long vowels under the consonant-vowel or the vowel-consonant scenarios, respectively, so, (33*12 = 396) (.WAV) files have been prepared for the training phase (twelve token per phoneme), and the same number of files for the testing phase.

Note that the recording processes is performed in a relatively quiet room; no special efforts were made in order to diminish whatever noise sources being introduced to this data at the moment of recording. Note also that the voice signals in this work are recorded by one female speaker.

2.2 Preprocessing

After recording process, both training and testing data should be segmented and preprocessed (i.e. filtered and scaled). Note that all the segmentation processes here have been handled manually, with different segmentation accuracies. That is, the training data have been more accurately segmented in comparison to the test data, which has been less accurately segmented, in order to give a better indication of the real world recognition results that mainly depend on automatic segmentation approaches, which are usually much less accurate in determining phoneme boundaries than humans.

The filtering process is handled using an 8-order band pass IIR filter. Next, the filtered signal of each phoneme is to be scaled to ensure that its amplitude vary over the entire range [-1, 1]. This can be done as follows:

Let X_{max}= max |X(n)|. Then

$$X_{scaled}(n) = \frac{X(n)}{X_{\max}} \quad n = 1, 2, 3, \ldots N. \qquad (1)$$

Where N : is the total number of samples.

Here, each sample of the phoneme signal *X* is divided by the absolute value of the sample that has the maximum amplitude within that signal.

2.3 Feature Extraction

Finding an efficient data representation has been a major concern in the field of pattern recognition. In this section, a hierarchical tree architecture based on MFCC features is proposed in order to recognize the phonemes belonging to the different classes. The prior knowledge of phoneme classes supported by the manner of articulation is partly used in modeling the tree in here.

2.4 Phoneme Recognition using MFCC- based Features

Here, the scaled phoneme signals are divided into non-overlapped frames 8 msec long (128 samples) each. A hamming window is then applied to these frames. 22 (see later) triangular-shaped half-overlapped filters are chosen to calculate the MFCC transform here, with theses filters being linearly spaced over the Mel frequency scale. This would give 22 cepstral parameters for each token (including the 0th coefficient). Although that the 0th coefficient of the MFCC cepstrum is ignored in many automatic speech recognition systems, due to its unreliability [7], *Zheng et al.* [8] have experimentally proved that better results would be achieved when the 0th coefficient is included, because it contains energy information of several different sub-bands of the whole frequency spectrum. The choice of the previous number of filters (and hence MFCC coefficients) was based on the calculation of the discrimination (in terms of Mean Square Error MSE) among phoneme vectors. The MFCC vectors has been calculated for a different number of filter banks (12, 14, 16, …, 28) and coefficients (12,14,16,…,28), then the number of filter banks and coefficients giving the maximum discrimination have been chosen here. The MFCC features are extracted in three different manners:

1. **Mid-frame based MFCC features:** the mid-frame of the speech signal is preferable because it is far from the coarticulation effects those do clearly appear in the terminal (starting and ending) frames as a result to the inaccuracy of the segmentation process. Here the MFCC vectors are calculated for the middle frame of each phoneme signal, and then the MSE is used to measure the discrimination among these vectors, the number of filter banks and coefficients those have given the maximum MSE are shown in table 1.
2. **Average-based MFCC features:** The average of the MFCC vectors of all the frames for each signal is calculated, and then the MSE measure is applied as is mentioned above. The average of the MFCC vectors gives an equal importance to all the frames in the phoneme signal including the terminal frames ignoring the coarticulation effects. The results are shown in table 2.
3. **Weighted-average-based MFCC features:** Unlike the previous case, where all the frames in the given signals are equally weighted, frames in the middle here are given higher weights than the terminal ones (Note that, weighting is applied to the MFCC vectors rather than frames here). This can be achieved by using a hamming window (for example) to weight the MFCC vectors of the corresponding frames constituting the phoneme signal. The MSE is then applied to calculate the discrimination measure. The corresponding results are shown in the table 3.

From tables 1-3, it is possible to see that the discrimination is at its maximum value when a 22-filter bank is used, and that the maximum discrimination using this filter bank is achieved when all the 22 coefficients are included. Having selected the suitable MFCC features, it is possible now to proceed to the core of the system: the neural-network based phoneme recognition system. This system is based on a hierarchy of Multilayer Perceptron MLP neural networks those are to divide the overall recognition task among the constituting networks. Here, the decisions from the distinct networks are combined in a hierarchical manner to form the overall network output.

The tree shown in Fig. 1 represents the classification of the Arabic phonemes according to the manner of articulation, while the proposed tree (shown in Fig. 2) is designed according to the

Table 1: Total Mean Square Error (MSE) of mid-frame MFCC for different numbers of filter banks and coefficients (coef.).

No. of coef.	Number of filter bank								
	12	14	16	18	20	22	24	26	28
12	1537300	2091.7	4579.5	6986.1	18447	23693	784.2	834.81	1826.4
14		16841000	2066.6	10479	23770	53929	738.39	562.01	924.91
16			1346300	17756	178940	1118000	817.67	515.89	644.64
18				8148300	112830	3839600	1009.2	586.16	820.25
20					3182500e+1	2207600e+1	906.59	781.03	996.92
22						1350200e+2	579.23	744.1	737.64
24							6785900	729.86	749.85
26								816480	649.03
28									751890

Table 2: Total Mean Square Error (MSE) of average-based MFCC for different number of filter banks and coefficients (coef.).

No. of coef.	Number of filter bank								
	12	14	16	18	20	22	24	26	28
12	541390	480.95	681.19	1091.5	2066.6	2246.9	259.39	286.16	529.84
14		848450	663.39	1923.7	3830.9	6909.9	313.12	246.6	385.05
16			579920	1446	2908.9	11420	244.66	195.02	234.27
18				9518900e+1	4873.2	12039	311.84	210.91	276.88
20					1382300e+2	6551.7	279.15	235.66	292.32
22						3198300	218.68	239.66	264.51
24							358710	248.14	261.65
26								281540	252.76
28									275860

Table 3: Total Mean Square Error (MSE) of weighted-average-based MFCC for different number of filter banks and coefficients (coef.).

No. of coef.	Number of filter bank								
	12	14	16	18	20	22	24	26	28
12	840830	721.45	1084.8	1709.9	4323.3	5395.5	387.71	429.32	799.57
14		2958800	894.45	2882.9	6286.5	13544	440.99	342	535.77
16			849330	8141.7	137290	52506	412.84	300.53	365.02
18				5040500	226510	85285	502.16	319.52	425.27
20					5700600	107190	464.77	375.54	472.57
22						1727500e+3	337.68	381.35	401.48
24							628580	398.56	407.82
26								410740	385.67
28									389590

prior knowledge about the phoneme classes, manner of articulation, and the requirements for the design of a HTR that has the smallest error rate, which have been empirically concluded. Each MLP in the tree receives 22 inputs, corresponding to the input features. Different numbers of hidden units are examined here to determine their effect on the recognition accuracy, and the most suitable number of them is chosen for each branch of the tree.

Here the MLP will be trained on speech data using the Resilient Back-Propagation (Rprop) algorithm [9, 10], where an enhanced performance can be achieved.

Note that in the proposed tree structure, the training data to be recognized by each node is also shared by all its direct successor nodes, with the amount of required training data being decreased as the sub-branches of the tree are traversed away from the root node (this due to the decrease in ambiguity resulting from decreasing the total number of candidate classes as these sub-branches are traversed). However, due to the effect of the error boosting through the successive sub-class recognition levels constituting the tree, the design of hierarchical structures becomes crucial.

At a given non-terminal node, the decision on which sub-branch is to be chosen depends on the outputs of the neural networks descending from that node, with the sub-branch corresponding to the neural network that has produced the minimum error being chosen (note that the error here is the difference between the output of the neural network and the predetermined output value). The recognition process completes when a decision is made to branch into a leaf (a terminal node), whose class is to correspond to that of the input data.

The recognition accuracies (rec. acc.) of the consonants and vowels classes for the proposed system are shown in the Table 4, where the class recognition accuracy of any phoneme can be given as in equation (2):

$$class \quad rec.acc, = \frac{Number \;\; of \;\; correctly \;\; recognized \;\; classes}{Total \;\; number \;\; of \;\; test \;\; phonemes} * 100\% \tag{2}$$

with the important case for the single-phoneme-wide class recognition accuracy being given by equation 3:

$$phoneme \quad \text{rec.acc} = \frac{Number \;\; of \;\; correctly \;\; recognized \;\; phonemes}{Total \;\; number \;\; of \;\; test \;\; phonemes} * 100\% \tag{3}$$

Vowels here branch into six leaves, for each one of them the recognition accuracy is tested against the number of neurons in the hidden layer to conclude the suitable structure that improves the performance of the tree.

However, the neural networks those have the highest recognition rate (for the corresponding phonemes or classes of phonemes) are not always selected here, neural networks those offer a better compromise between the recognition rate and the error induced by these networks into their surrounding networks are sometimes selected.

From the general tree topology in Fig. 1, consonants are classified here into voiced and unvoiced classes, which can be also recognized using this method.

Table 4: Recognition accuracy for the consonants-vowels neural net.

Network name	Recognition accuracy %

consonants-vowels	Consonant = 98.45
	Vowels = 93.05

The recognition accuracies of the voiced and unvoiced classes are shown in the Table 5. However, in order to reduce the effect of error boosting this level has been eliminated from the proposed tree; so, the consonants here are divided directly into 11 groups based on criteria those will be explained.

The elimination of voiced-unvoiced decision level from the tree causes the stop and fricative classes to have many phonemes those differ in the manner of articulation (voiced and unvoiced), which would decrease the capability of the corresponding networks to discriminate among them, hence, reducing the recognition accuracy of these phonemes. Therefore, the stop and fricative classes have been partitioned into stop voiced, and stop unvoiced, fricative voiced and fricative unvoiced subclasses, respectively.

Table 5: Recognition accuracy for the voiced-unvoiced neural net.

Network name	Recognition accuracy %

voiced-unvoiced	Voiced = 82.73
	Unvoiced = 78.2

The recognition accuracy of the fricative-voiced class was very poor (below 40%), especially for the / γ / and / ʕ / phonemes. One way that is proposed to improve the final recognition accuracy is to partition this subclass into groups according to the MSE measured among the MFCC vectors of the phoneme signals in the same class. Phonemes those have high MSE with respect to the remaining phonemes in this class are put into separate groups. The MSE between / ʕ / phoneme and the stop-unvoiced phonemes is less than that between it and the phonemes in its class. So, joining the / ʕ / phoneme with the stop-unvoiced class would increase the recognition accuracy of that phoneme without affecting the overall accuracy of the other phonemes.

The recognition accuracy of the stop-unvoiced class is equal to 90%. However, as the / ʕ / phoneme is joined to the stop-unvoiced class the recognition accuracy reaches to 94.44%.

On the other hand the MSEs among the / s /, / ʃ /, and / s̱ / phonemes and the / ħ /, / x /, / θ /, / f / and / h / phonemes class in the fricative-unvoiced are relatively high. So, partitioning the fricative-unvoiced class into two groups, one consisting of / s /, / ʃ /, and / s̱ / phonemes (that would be called fricative-unvoiced1), and the other consisting of the / ħ /, / x /, / θ /, / f / and / h / phonemes (that would be called fricative-unvoiced2 here) would improve these subclasses and the overall recognition accuracy.

The MSE between / z / phoneme and the / s /, / ʃ /, and / s̱ / phonemes in the fricative-unvoiced1 class is less than that between it and the phonemes in its original class. So, joining the /z / phoneme with the fricative-unvoiced1 class would increase the recognition accuracy of that phoneme from 91.66% to 100% without affecting the overall accuracy of the other phonemes. Moreover the recognition accuracy of the fricative-unvoiced1 class has been increased from 97.22% to 100%.

After the phonemes / ʕ / and / z / are split away from their original class, the fricative-voiced class would contain / γ / and / ð / phonemes only, which produced a poor recognition accuracy when they are kept together, so they are partitioned into two groups each containing one phoneme only.

In the next level of the tree, each one of the subsets is branched into terminal nodes those represent the phonemes in that subset. Having determined the structure of the hierarchical tree recognizer, the recognition accuracy of each phoneme can be shown in Table 6.

The flat MFCC recognizer is also tested here to reveal the advantages of the proposed hierarchical system in comparison to non-hierarchical (flat) systems. One-Class-One-Network (OCON) architecture is used for the flat recognizer here that is a separate network is trained for each phoneme.

The recognition accuracy of each phoneme can be shown in Table 6, too. The class recognition accuracies for the hierarchical and flat recognizer system are shown in Table 7.

3. CONCLUSIONS

In this paper a hierarchical Arabic phonemes recognition system has been built that used 22 Mel frequency cepstral coefficients, which represent a short term energy spectrum expressed on a Mel–frequency scale. A different number of MFCC coefficients for a different number of filter banks have been tested here, and in three different manners: the mid frame based, average based, and weighted average based MFCC coefficients. The 22 coefficients-22 filter bank MFCC features (which have revealed the maximum discrimination among phoneme signals) were selected as the input features to the hierarchical system.

The "hard split" of data has been used in modeling the tree structure, that is dividing the problem into sub-problems those have no common elements. However, as it is practically impossible to find an optimal structure by an exhaustive search through the MSE among the MFCC vectors (the classification problem), evidences from MSE data and the tree structure design heuristics have been applied.

By the aid of MSE calculations among the MFCC vectors of the phoneme signals some phonemes have been grouped into new classes. Phonemes those have high MSE with respect to the other phonemes in their original class are put into other groups, thus the stop-unvoiced+/ ʕ /, fricative-unvoiced1, fricative-unvoiced2+/ z /, /ð/, and /γ/ new classes are presented here. Due to the error boosting from level to level in the tree, the recognition accuracy is degraded sometimes. However, it is observed that the hierarchical system still performs significantly better than the corresponding monolithic recognizer, with a recognition accuracy of 68.181%, compared to the flat system recognition accuracy 51.262%.

4. References

[1] Ali A. A., Alwan M. A. and Jasim A. A., "*Hybrid Wavelet-Neural /FFT-Neural phoneme recognition*". The second International Conference on Information Technology, Al-Zaytoonah University of Jordan Faculty of Science & Information Technology, PP.39-47, 2005.

[2] Siva Rama Krishna Rao J. Y. "*Recognition of Consonant –Vowel (CV) Utterance Using Modular Neural Network Models*", Msc. Thesis, Department of Computer Science and Engineering, Indian institute of Technology, Madras, 2000.

[3] Tan Lee "*Automatic Recognition of Isolated Cantonese Syllable Using Neural Network.*", Ph.D. Thesis, Department of Electronic Engineering, University of Hong Hong, 1996.

[4] Fritsch J. and Finke M. "*Applying Divide and Conquer to Large Scale Pattern Recognition Tasks.*", Interactive Systems Laboratories, 1996.

[5] Jordan M. I. and Jacobs R. A. "*Hierarchical Mixtures of Experts and EM Algorithm.*", Neural Computation, Vol.6, PP.181-214, 1994.

[6] Safavian S. R. and Landgrebe D "*A Survey of Decision Tree Classifier Methodology*", IEEE Transactions on Systems, Man, and Cybernetics, Vol. 21, No.3, PP. 660-674, 1991.

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

[8] Zheng F., Zhang G., and Song Z., "*Comparison of Different Implementation of MFCC.*", J. Computer Science & Technology, Vol.16, No. 6, PP. 582-589, 2001.

[9] MacKay D. J. C., "*Information Theory, Inference, and Learning Algorithms*", Cambridge University Press, 2004.

[10] Riedmiller M. and Braun H. "*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.

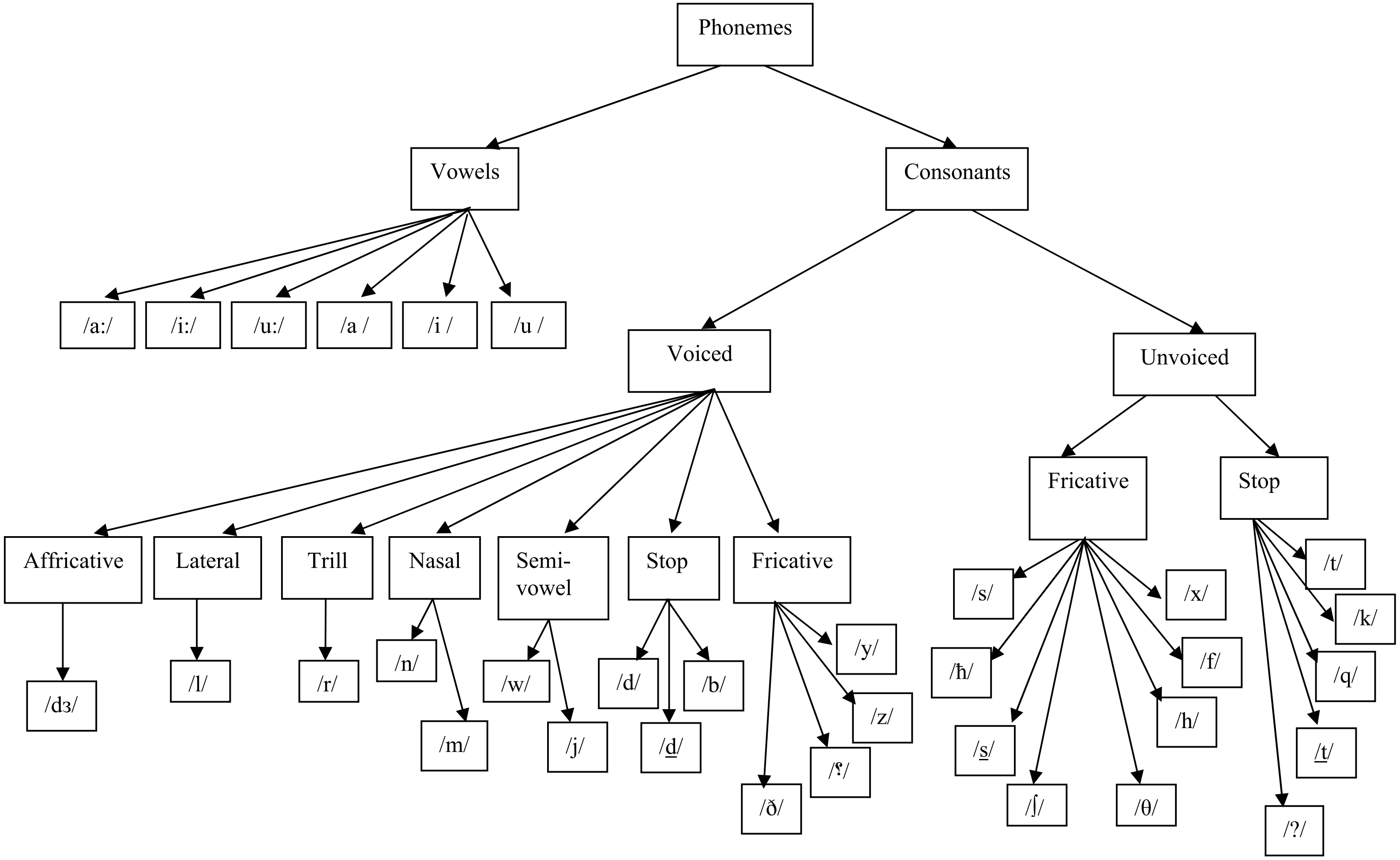

Fig. 1: The classification of Arabic phonemes

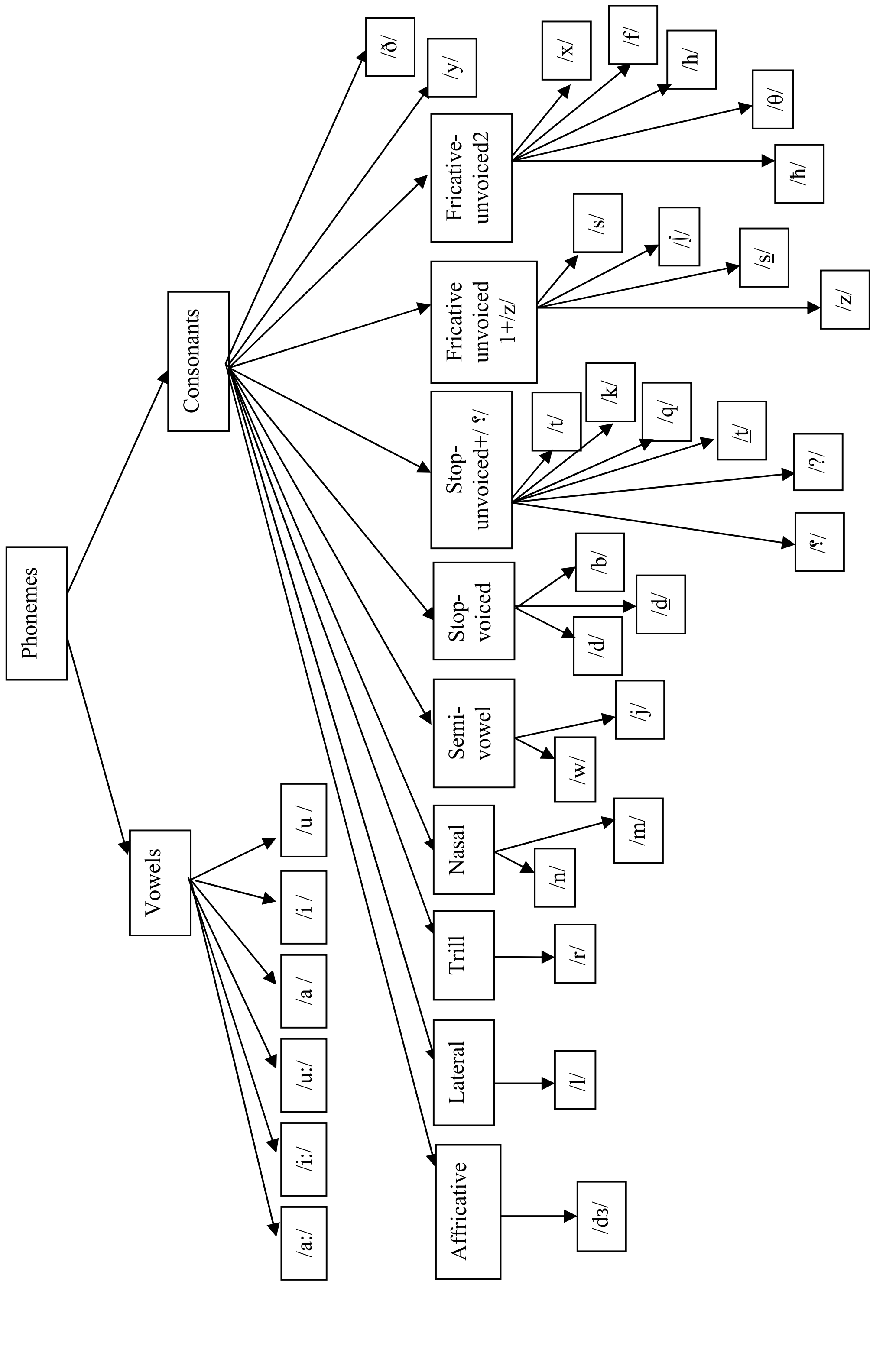

Fig. 2: The proposed phonemes recognition tree

Table 6: Hierarchical vs. flat MFCC- neural phoneme recognition accuracy.

phoneme	IPA symbols	Hierarchical phoneme recognition accuracy %	Flat phoneme recognition accuracy %
ء	/ ? /	75	50
آ (حرف علة)	/ a: /	100	75
ب	/ b /	58.33	66.66
ت	/ t /	75	50
ث	/ θ /	58.33	50
ج	/ dʒ /	75	50
ح	/ ħ /	83.33	83.33
خ	/ x /	66.66	66.66
د	/ d /	58.33	33.33
ذ	/ ð /	66.66	33.33
ر	/ r /	66.66	66.66
ز	/ z /	100	75
س	/ s /	50	50
ش	/ ∫ /	100	75
ص	/ s̲ /	66.66	50
ض	/ d̲ /	16.66	8.33
ط	/ t̲ /	66.66	41.66
ع	/ ʕ /	75	50
غ	/ γ /	50	25
ف	/ f /	66.66	75
ق	/ q /	50	25
ك	/ k /	58.33	33.33
ل	/ l /	75	58.33
م	/ m /	66.66	50
ن	/ n /	33.33	41.66
هـ	/ h /	58.33	33.33
و	/ w /	83.33	50
ي	/ j /	100	50
ي (حرف علة)	/ i: /	91.66	75
و (حرف علة)	/ u: /	41.66	33.33
فتحة	/ a /	83.33	75
كسرة	/ i /	58.33	41.66
ضمة	/ u /	75	50
Total		68.181	51.262

Table 7: Hierarchical vs. flat MFCC- neural class recognition accuracy.

class	Hierarchical class recognition accuracy %	Flat class recognition accuracy %
voiced fricative	75	47.91
Unvoiced fricative	91.66	85.41
voiced stop	66.66	58.33
Unvoiced stop	86.66	55
Nasal	70.83	70.83
Affricative	75	83.33
Semivowels	91.66	58.33
Trill	66.66	66.66
Lateral	75	58.33
Total	77.68	64.9