Phonemes Recognition of Speech Using Breeder Genetic Algorithm and Genetic Algorithm

Samaher Hussein Ali[1], Eman S. Al-Shamery[2]

[1]*Department of Computer Science, University of Babylon, Iraq*

Samaher_hussein@yahoo.com

[2] *Department of Computer Science, University of Babylon, Iraq*

emanalshamery@yahoo.com

Abstract: phoneme recognition can be viewed as classifying multivariate observation. Breeder Genetic Algorithm (BGA) and Genetic Algorithm (GA) approach the decision problem using two complementary models. For technical speech recognition systems as well as for humans it has been shown that the combination of acoustic and optical information can enhance speech recognition performance. In this paper we systematically investigate the phoneme recognition problem, where phones are a useful representation because words can easily be-written as phones using a lexicon. The propose models depend on three stages: the first stage is preprocessing of speech by token speech into phonemes, the second stage includes selection phonological features over phones (i.e. cepstral coefficients), the final stage includes using the Evoluationary Algorithm (BGA & GA) to clustering the dataset and recognition each phonemes.

Key words: Breeder Genetic Algorithm, Cepstral Coefficients, Genetic Algorithm, Phoneme Recognition.

1. INTRODUCTION

Current speech recognition systems are usually based on Hidden Markov Models (HMMs). Recently, research has also focused on alternative ways for modeling the speech signal [1].

Most speech recognisers today are based on phones (or phonemes) which, in our opinion, are often given undue legitimacy in the speech community, particularly with respect to the assumption that a sequence of acoustic observations can be synchronized with a sequence of phones. Often phones are seen as being the "atoms" of speech in that they are the set of units from which all else (that is, word sequences) can be built. But just as with atoms in physics, it is now widely accepted in phonology that phones are decomposable into smaller, more fundamental units. There is no consensus as to what these units are, but the most popular view is that phones can be constructed from a set of phonological distinctive features.

Phoneme recognition can be viewed as classifying multivariate observation. Phones are a useful representation because words can easily be re-written as phones using a lexicon. In conventional HMM systems, phones are then re-written as HMM states. HMMs are generative models, with each observation generated by a single state. During recognition, the state sequence is hidden, and the probability that a particular model has generated a given sequence of observations is often calculated approximately using Viterbi decoding. We argue here that it is inappropriate to align observations, phones and words in this strict fashion. In HMM speech recognition, the acoustic model which relates states to observations effectively does two jobs: 1) it turns representations from the acoustic domain to a phonetic one; 2) it models the time dynamics of the acoustics so that sequences of similar acoustic observations are modeled as single phone. Our feature based approach performs each of these operations separately. The feature extraction component maps from the acoustic domain to the phonetic domain, but the representation is still one of frame based time varying vectors. The phone model then operates in the feature domain. The phonological features could be regarded as latent variables: a simple, hidden process which gives rise to a (more complex) observable process. The basic problem with performing the two steps at once is that the HMM has no inherent ability to model the dynamics of the acoustic observations[1]. Speech production from phones to acoustics is complex and non-linear and hence phenomena which can have relatively simple phonetic explanations can give rise to extremely complex acoustic patterns. While it is easy to model nasality spread phonetically, it is very difficult to do so in the acoustic domain as the effects of nasality can not be represented by a simple function operating on the acoustics.

[1] The combination of a discrete state and the Markov property restricts trajectories of parameter means to be piecewise constant. This situation is mitigated a little by appending first and second differences to the observation.

The remained of this paper is structure as following section two: focuses on the theoretical concepts related to the Evoluationary Algorithm: selection, fitness function, crossover, mutation, elitism concept. Section three: explain the proposed system used in Recognition Phone for speech. Each step in the proposed system has been explained and analyzed extensively.Section four: illustrates the implementation of proposed system and the results of the cases study. Section five: show conclusions of this work together with some recommendations for future work in this field

2. EVOLUATIONARY ALGORITHM

The term EA refers to a big family of search methods based on concepts taken from Darwinian evolution of species and natural selection of the fittest [2]. Moreover, EA uses the Darwinian principle of "survival of the fittest" to evolve optimum solutions to problems [3]. Additionally, it maintains a population of individuals that represent potential solutions to it. The three main representatives of EAs are [2][3]: Genetic Algorithms (GAs) proposed by Holland 1975, Evolution strategies (ES) developed by Rechenberg and Schwefel during 60's and more or less settled in 70's, Evolutionary Programming (EP) introduced by Fogel.

EA maintains a population of individuals that represent potential solutions to it [3]. Each individual in the population is represented by chromosome consisting of a string of atomic elements called genes. Each gene represents a variable, either for the problem or for algorithm itself. The possible value of a gene is called alleles and gene's position in the chromosome is called locus (loci).

There is also distinction between the genotype, (i.e., the genetic material of an individual) and the phenotype, (i.e., the individual result of genotype development). In EAs the genotype coincides with the chromosome, and the phenotype is simulated via a fitness function, a scalar value-similar to a reinforcement expressing how well and individual has come out of a given genotype.

The search process usually starts with a randomly generated population and evolves over time in a quest for better and better individual where, from generation to generation, new populations are formed by application of three fundamental kind of operators to the individuals of a population, forming a characteristic a three step procedure[2]:-

1- Selection of the fittest individuals, yielding the so-called gene pool.

2- Recombination/crossover of the previously selected individuals forming the gene pool, giving rise to an offspring of new individuals.

3- Mutation of the newly created individuals. By iterating this three-step mechanism. It is hoped that increasingly better individuals will be found. This reasoning is based on the following ideas [4]:-

1- The selection of the fittest individuals ensures that only the best ones will be allowed to have offspring, driving the search towards good solutions.

2- By recombining the genetic material of these selected individuals. The possibility of obtaining an offspring where at least one child is better than any of its parents is high.

3- Mutation is mean to introduce new rails, not present in any of the parents. It is usually performed on freshly obtained individuals by slightly altering some of their genetic material.

4- Replacement criterion is the last operation that basically says which elements, among those in the current gene-pool and their newly generated offspring, are to be given a chance of survival on to the next generation. And there are two basic strategies of replacement (The plus strategy, the comma strategy)

Clustering [5] is a popular unsupervised pattern classification technique which partitions the input space into K regions based on some similarity/dissimilarity metric. The number of partitions/clusters may or may not be known a priori.

In general the elements of clustering techniques include [6]: Pattern representation (i.e., determines number of clusters, number of variable vectors and number of features in the feature vector), Feature selection (i.e., defines a subset of features to use in clustering process), Data abstraction (i.e., represents a process to find simple representation of clustering sets), Assignment measure (i.e., explains how we can combine feature vectors by feature selection of one of the variable cluster). There are two types of these measures

a. Distance measures such as Euclidean distance, Minkowski distance.

b. Similarity measures such as Vector inner product.

Genetic algorithms (GAs) belong to a class of search techniques that mimic the principles of natural selection to develop solutions of large optimization problems.

GAs operates by maintaining and manipulating a population of potential solutions called chromosomes. Each chromosome has an associated fitness value which is a qualitative measure of the goodness of the solution encoded in it. This fitness value is used to guide the stochastic selection of chromosomes which are then used to generate new candidate solutions through crossover and mutation. Crossover generates

new chromosomes by combining sections of two or more selected parents. Mutation acts by randomly selecting genes which are then altered; thereby preventing suboptimal solutions from persisting and increases diversity in the population. The process of selection, crossover and mutation continues for a fixed number of generations or until a termination condition is satisfied [7].

Breeder Genetic Algorithm (BGA) represents a class of random optimization techniques gleaned from the science of population genetics, which have proved their ability to solve hard optimization problems with continuous parameters. BGA which can be seen as a recombination between Evoluation strategies (ES) and Genetic Algorithm (GA), uses truncation selection and the search process is mainly driven by recombination making BGA very similar to GA. It has been proven that BGA can solve problems more efficiently than GA due to the theoretical faster convergence to the optimum and they can, like GA, be easily written in a parallel form [8].

3. THE PROPOSED METHODOLOGY

The objective of this paper is to present a method to systematically investigate the phoneme recognition problem, where phones are a useful representation because words can easily be-written as phones using a lexicon. The propose models depend on three stages: the first stage is preprocessing of speech by token speech signals into phonemes, the second stage includes selection phonological features over phones (i.e. cepstral coefficients), the final stage includes using the Evoluationary Algorithm (BGA & GA) to clustering the dataset and recognition each phonemes. Figure 1 explains the block diagram of system. The different steps of this method are now discussed in detail.

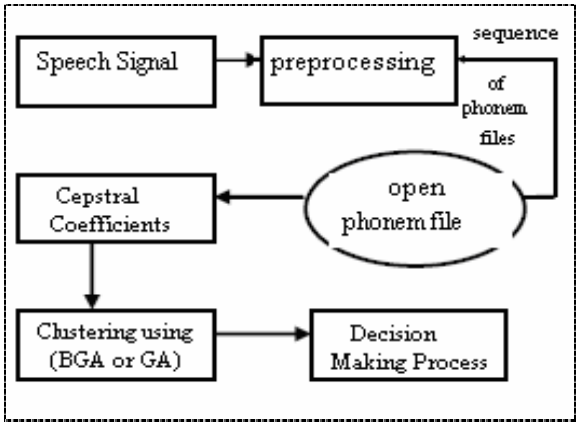

Fig. 1: the Structure of the proposed method

3.1. Preprocessing

The material has been recorded using cool edit pro by 11025 sampling frequency save in file wave. The preprocessing represent the silences elimination from start and end the wave file then divide it's into sub files each one is phoneme file manually.

3.2. Cepstral Coefficients

There are several methods for spectral analysis, which are in time or in frequency domain .the choice of method, depend on the implementation type; the famous methods are filter bank and the cepstral analysis. The cepstral analysis It is used for recognizing between a voiced sounds (v) and an unvoiced sound (uv). The cepstrum of voiced sound contributes a strong peak in the range 3-10 msec while this peak is absent in the cepstrum of unvoiced sounds. Thus the cepstral is used to deciding whether an input signal v or uv if its v ,the pitch period is measured from location of the cepstral peak ,this value is compared with the pre-set threshold value which derived empirically ,and the sound is v if it is cepstral peak is above the threshold value but if it's below ,this is not necessary uv .therefore ,a helpful factor in v/uv decision is the existence of a very strong positive peak below 0.5 msec of the v cepstrum ,while for uv cepstrum a strong negative peak exists below 0.5 msec

There are several methods to get of there coefficients ,one of these method depending on Fast Fourier Transform (FFT) to achieve this analysis ,should be implement these steps

1. Compute Discrete Fourier Transform (DFT) of speech waveform.
2. The logarithm of the transform is calculated.
3. The inverse DFT (IDFT) of the transform is obtained.

The cepstral coefficients can be directly calculated using signal processing on a frame by frame basis from the speech waveform. And we represent the window samples which applied to speech signal for each frame. There are many types of window; the hamming window is applied in this search because their forms appropriate with the signal form as show in the following equation [9]:

$$w(m) = 0.54 + 0.64\cos(2\pi m / N1 - 1) \quad (1)$$

Where m: sample index, W: hamming window , N1: length of window(256)

The speech signal is windowed by 256 point hamming window. The other method depending linear predictive coding technique (LPC), The product signal is estimated from the combination of the past speech samples according to the following equation [10, 11]:

$$sp\,(n\hat{)} = \sum_{k=1}^{p} Lpc_{k}\; sp\,(n - k) \quad (2)$$

Where LPC: the LPC coefficients, Sp: the actual signal, $s\hat{p}$: the product signal.

Cepstral Coefficient: The LPC parameters are considered very important which it is play main role in the speech processing .these coefficients are extracted depending on the LPC coefficients according to the following equations:

$$Cep_{0} = \ln G^{2} \quad (3)$$

$$Cep_{m} = Lpc_{m} + \sum_{k=1}^{m-1} (k/m).Cep_{k}.Lpc_{m-k} \quad (4)$$

$$Cep_m = \sum_{k=1}^{m-1} (k/m).Cep_k.Lpc_{m-k} \quad (5)$$

Where

Cep :(Cepstral Coefficients).

m: index of order of coefficients (i.e., in this work, we use m =18).

3.3. Clustering using (BGA or GA)

In this section, an attempt has been made to use breeder genetic algorithms or genetic algorithm for automatically clustering dataset and recognition each phonemes. This includes determination of number of clusters as well as appropriate clustering of the data. The methodology is explained first followed by the description of the implementation results.

3.3.1. Representation of Solution

The chromosomes are make up of real numbers (representing the cepstral coefficients drawing from phone dataset). The length of a string is taken to be K_{max} where each individual gene position represents one of the cepstral coefficients. In this work, we use 18 coefficients therefore each individual consist of 18 genes.

3.3.2. Population Initialization

For each string i in the population (i=1,2,…, p, where p is the size of the population), a random number K_{max} is generated. This string is assumed to encode the centres of clusters. For initializing these centres, K_{max} points are chosen randomly from the dataset of features. These points are distributed randomly in the chromosome.

3.3.3 Fitness Function

The fitness of a chromosome is computed using the average Euclidean distance of the vectors in class k to the centroid of class k, is computed as

$$d(X, Z^{(i)}) = \sum_{k=1}^{K} w_k \left| \frac{(X_k - Z_k^{(i)})}{r_k} \right| \quad (6)$$

Where, k=1,2,…,K, x is feature vector, $Z^{(i)}$ is center of cluster i , k number of features in feature vector (in this work k=18) , w_k is weight of feature k ($0<w_k<1$), r_k is the range of feature k.

The objective is to minimize the $d(x,z^{(i)})$ for achieving proper clustering. The fitness function for chromosome j is defined as $1/ dj(x,z^{(i)})$, where dj is the Euclidean distance computed for this chromosome, where the maximization of the fitness function will ensure minimization of the Euclidean distance.

3.3.4 Evoluation Operations

The following Evoluation operations (for genetic & breeder genetic algorithm) are performed on the population of strings for a number of generations.

Selection

In genetic algorithm, we use, the "roulette wheel" selection is a process in which a subject representing x % of total fitness has x % chances to be selected for mating. Since a same subject can be selected twice during one cycle, it's both father and mother, and therefore if there is no mutation, the offspring won't present any difference from its parent [12].

In breeder genetic algorithm, we used, truncation selection is artificial selection method in which only the best individuals-usually a fixed percentage of total population size p are selected and the gene pool to be recombined and mutated is entered . As the basis to form a new generation, usually truncation ratio lies in rang [10 %, 50%]. The BGA selection mechanism is then deterministic (i.e., there are no probabilities), extinctive (the best elements are guaranteed to be selected and the worst are guaranteed not to be selected).And 1-elitist (the best element is always to survive from generation to generation).

Recombination

Any operator combining the genetic material of the parents is called a recombination or crossover operator.

In GAs crossover is applied conditionally, in this work we used uniform crossover (ux). UX crossover creates offspring by deciding for each allele of one parent, whether to swap that allele with the corresponding allele in the other parent ($p_c \approx 0.5$)

In BGAs recombination is applied unconditionally. let $\vec{A}$=(a_1,a_2,…..,a_n), $\vec{y}$=(a_1,a_2,…..,a_n) be two selected gene-pool individual $\vec{a}$, $\vec{b}$ such that $\vec{a} \neq \vec{b}$ .Let $\vec{c}$=(c_1,c_2,…..,c_n)be the result of recombination and 1≤i≤n . The following are some of the more common possibilities to obtain an offspring. During recombination each cluster centre is considered to be an indivisible gene. three types of recombination process are discussed in this work:-

A. Discrete Recombination (DR)

$$C_i \in \{a_i, b_i\} \quad (7)$$

Chosen with equal probability

B. Extended Line Recombination (ELR)

$C_i = a_i + \alpha(b_i - a_i)$, where $b_i \geq a_i$ (8a)

$C_i = b_i + \alpha(a_i - b_i)$, where $b_i < a_i$ (8b)

With α uniformly random chosen in [-d,1.0+d] where d is a parameter for the BGA and d ≥ 0 (typical d=0.25).

C. Extended Intermediate Recombination (EIR)

$C_i=a_i+\alpha_i(b_i-a_i)$, where $b_i \geq a_i$ (9a)

$C_i=b_i+\alpha_i(a_i-b_i)$, where $b_i < a_i$ (9b)

With αi uniformly random chosen in [-d,1.0+d] the difference with ELR being that in this latter case we choose a new α_i for each i.

Mutation

Mutation changes a gene in the chromosome with small probability. In GA, mutation is a mechanism for restoring lost and unexplored genetic material into the population and searching regions of the allele space not generated by selection and crossover. It can be used to prevent premature conversion; this increases the divisibility of genetic material. In this work, we use (change the value of gene) as mutation method of GA.

While in BGA, each position in a chromosome is mutated with probability pm=1/n so that, on average, one gene is mutated for each individual, as follows:

$$ci'=c_i \pm searchinterval_i .Const. \sum_{j=0}^{k-1} Qi \ . 2^{-j} \quad (10)$$

In the above formula k is a parameter originally related to the machine precision, (i.e., the number of bits used to represent a real variable in the machine). We are working with (e.g. 24, 32, and 64). And const determines the maximum half–width of the interval centered in ci in which c_i' can be. In this work the searchinterval is represented (number of features* number of phonemes) of the dataset and const represented half-searchinterval of dataset. Furthermore each Qi equal (zero) before mutation and is mutated to (one) with probability (1/k), so on average just one of the elements in the sum with be non-zero after mutation. A practical assumption is that we deterministically flip just one and only one of these bits, so that the above formula becomes:-

$$c_i'=c_i \pm searchinterval_i \ . \ const. \ 2^{-j} \quad (11)$$

Where $0 \leq j \leq k-1$

3.4. Decision Making Process

After verification of one of the stopping criterion to Evoluation algorithm(GA, BGA) such as verified cost function condition or exceeding the number of generation to maximum number of generation, we can say that the GA or BGA is complete there work.

The best string having the largest fitness (i.e., smallest Euclidean distance value) see up the last generation provides the solution to the clusters count problem. We have up implanted elitism at each generation by preserving the best string see up to that generation in a location outside the population. Thus on termination, this location contains the centers of the final clusters that represent the dataset after clustering process and also provide recognition of phoneme.

4. IMPLEMENTATION AND RESULTS

Two Evoluation algorithms have been trained to recognise phonemes in continuous speech. The material was recorded by two male Iraqi speakers and two female women Iraqi speakers. The sentences were phonetically labeled. The labeling was basically phonemic and did not show allophonic variations. Forty sentences were used for recognise. The number of phonemes for the clustering material was 2202 and the total number of 10 ms frames was 15258.

Result with the GA, for the GA the number of individuals in the population has been fixed equal to 50 as termination criterium we have established a value of 100 for the maximum number of generations. It utilizes the roulette wheel selection with a 1-elitist strategy. The operators are the uniform crossover (ux) with p_c=0.9 and the mutation with p_m=0.04. Table (1) explained the other details.

Results with BGA, the successive step we has taken consists in the utilization of non-binary technique, BGA able to directly deal with the integer or real variables involved in the problem under examination.

The number of individuals in the population and termination criterion for the BGA has been the same used for the GA. It utilizes the truncation selection with a 1-elitist strategy and the percentage of the truncation equal to 30%. The discrete mutation with p_m=0.04 has been employed.

In particular, the results for the BGA have improved in terms of convergence speed by using the extended intermediate and the extended line recombination instead of discrete recombination operators.

Table 1: Summary of GA and BGA Results

Type of Recombination	P_c	P_m	NO. Of Generation	NO. Of Clusters	Fitness Value
UX	**0.8**	**0.04**	**100**	**15**	0.62167
DR	**0.9**	**0.04**	**87**	**12**	0.51183
ELR	**0.9**	**0.04**	**51**	**23**	0.85831
EIR	0.9	0.04	10	35	0.91120

5. CONCLUSTION

The primary aim of this paper has been to describe

techniques for the recognition of phonemes from continuous speech depends on phonological features. However, it is now worth discussing some issues concerned with actual *recognition* that is the conversion of feature descriptions for an utterance into linguistic units such as phones or words. The long term goal is to develop or adapt statistical models which make explicit use of the benefits of features, for example by assuming conditional independence between the different feature values in a frame, and by modelling co-articulation with reference to the theory of critical articulators. While this is the subject of current and future work, it certainly is reasonable to ask at this point what evidence we have that we are on the right track and that we haven't simply developed an interesting representation that will prove of no great benefit towards solving the larger problem.

Our results show that it is essential to capture co _articulation information when doing phoneme recognition. One way to include this is by expanding the size of the input spectral window (number of features: spectral coefficients) used by the EA. This was shown to significantly improve performance for both tested models. The experimental results have, firstly, proved the effectiveness of EA for the optimization. Then, they have proved the superiority of the BGA with respect of GA in terms of both solution quality and speed of convergence particular.

REFERENCES

[1] Freitag. F, "Acoustic-Phonetic Decoding Base on Elman Predictive Neural Networks", Universitat Politècnica de Catalunya, Department of Signal Theory and Communications, Barcelona, Spain, 1995.

[2] Lluis.B, "A Study in Function Optimization with the Breeder GeneticAlgorithm", 1999

Site:http://citeseer.ist.pus\cache\paper\cs

[3] Angela. B, "Soft Computing", England, 2003.

Site:http://www.tessella.com

[4] Lluis.A,"A Case Study in Neural Network Training with Breeder Genetic Algorithm", Spain, 2001.
Site:http://citeseer.ist.psu.edu/cache/papers/cs/14387/http:zSzzSzwww.lsi.upc.eszSz~belanchezSzrecercazSzPIMA.pdf/a-case-study-in.pdf

[5] Sanghamitra.B and Ujjwal.M, "Genetic Clustering For Automatic Evolution of Clusters and Application to Image Classification", the Journal of the Pattern Recognition Socirty, Vol.35, PP. 1197-1208, 2002.

Site:http://www.elsevier.com\locate\patcog

[6] Ahmed. K, "A Genetic Clustering For Image Segmentation ",M.Sc. Thesis, Babylon University, 2002.

[7] Tawfiq. A, Samaher Hussein and Israa Hadi, "Object Oriented Classification of Forest Images Using Soft Computing Approach ", the journal of Babylon university , Vol 15, No 3, 2006.

[8] Falco.I, Cioppa.A, Balio.R and Tarantino.E, "Investigating A Parallel Breeder Genetic Algorithm on The Inverse Aerodynamic Design", Naples-Italy, 2000.

Site:http://citeseer.ist.psu.edu/cache/papers/cs/1832/http:zSzzSzwww.irsip.na.cnr.itzSz~hotgSzpaperszSzppsn.pdf/defalco96investigating.pdf

[9] Flangan J.L,"*Speech Analysis"*, Springer_Verlag,New York, Second Edition,1972.

[10] Margin_Chagnolleau. M, Wike. N.J and Bimbot.F, "proc.ICASSP 96", IEEE, 15, 401, 1996.

[11] Rabiner. L and Juang. B.J, "*Fundamentals of Speech Recognition"*, prentice_ Hall. Ptr., Englewood Cliffs., New Jersey,1993

[12] Jean-Philippe R, "Genetic Algorithm Viewer: Demonstration of a Genetic Algorithm", May 2000

Site: http://www.rennard.org/alife , alife@rennard.org