Arabic Phonemes Transcription using Data Driven Approach

Khalid Nahar[1], Husni Al-Muhtaseb[1], Wasfi Al-Khatib[1], Moustafa Elshafei[1], and Mansour Alghamdi[2]
[1]College of Computer Science and Engineering, King Fahd University of
Petroleum and Minerals, Saudi Arabia
[2]Computer Research Institute, King Abdulaziz City for Science and Technology, Saudi Arabia

Abstract: *The efficiency and correctness of continuous Arabic Speech Recognition Systems (ARS) hinge on the accuracy of the language phoneme set. The main goal of this research is to recognize and transcribe Arabic phonemes using a data-driven approach. We used the Hidden Markov Toolkit (HTK) to develop a phoneme recognizer, carrying out several experiments with different parameters, such as varying number of Hidden Markov Model (HMM) states and Gaussian mixtures to model the Arabic phonemes and find the best configuration. We used a corpus consisting of about 4000 files, representing 5 recorded hours of Modern Standard Arabic (MSA) of TV-News. A statistical analysis for the phonemes length, frequency and mode was carried out, in order to determine the best number of states necessary to represent each phoneme. Phoneme recognition accuracy of 56.79% was reached without using a language model. The recognition accuracy increased to 96.3% upon using a bigram language model.*

Keywords: *Phoneme transcription, data-driven, speech recognition, network lattices, arabic speech corpus.*

Received November 20, 2012; accepted August 15, 2013; published online June 26, 2014

1. Introduction

Phoneme transcription plays an important role in the process of speech recognition, text to speech applications and speech database construction [7]. Two traditional methods are usually used for phoneme transcription; the feature input method, which is carried out by the speech recognition task and the text input method, which is carried out by Grapheme to Phoneme task (G2P) [10]. The maximum accuracy reached in continuous speech recognition with large vocabulary was 80% [10]. G2P gives more accurate recognition, but relies on a perfect pronunciation lexicon.

Phonemes recognition in continuous speech is not an easy task due to co-articulation and the inherent variability in the pronunciation of some phonemes. Stop consonants are short duration phonemes that are usually misclassified. In order to, reduce the error rate and the phoneme confusion, some additional information, such as the bigram model and the phonemes duration, are added to the recognizer during the recognition process [10].

State of the art continuous Arabic Speech Recognition (ASR) systems recognize Modern Standard Arabic (MSA) that usually appears in TV-News, newspapers and books. MSA is the formal style of writing and speech across the Arab world [3].

Arabic dialects and accents, however, pose great challenges in building Automatic Speech Recognition Systems (ASR). For example, MSA based ASR systems suffer from high Word Error Rate (WER) when applied to those dialects and accents. This study examines automatic data driven based transcription of the Arabic phonemes by developing a more suitable phoneme recognition process. This is accomplished through investigating the possible use of different number of Hidden Markov Model (HMM) states and different number of Gaussian mixtures, on the accuracy and speed of recognizing each phoneme. Multiple experiments were conducted with different parameters, using the Hidden Markov Toolkit (HTK). Four parameters were used in the experiments; the number of HMM states for each phoneme, the feature vector type and length, the existence or absence of a language model and the number of Gaussian mixtures for each phoneme.

The remainder of this paper is organized as follows: Section 2 describes the problem statement. Section 3 gives an overview of related work to this research. Section 4 details the research methodology and our experimental setup. Section 5 presents the results of our experimentation. Finally, we conclude the paper in section 6 and provide further future work.

2. Problem Statement

Many research work efforts have been conducted in continuous ASR using HMM with varying rates of accuracies. The recognition accuracy in our caseis measured by the percent of correctly recognized phonemes. The ASR accuracy is usually affected by several factors, including the phoneme set used, the number of HMM states for each phoneme and number of Gaussian mixtures.

Using efficient Arabic phonemes transcription in continuous ASR before using the language model increases the recognition speed and reduces the memory consumption caused by the recognizer. In addition, determining the number of HMM states to model each phoneme increases the accuracy of the recognition. It may also reduce the HMM chain for each utterance. The reason is that some phonemes need only one HMM emitting state to be modeled, while others may need two or more emitting states.

In this paper we propose to build a phoneme recognizer based on a data driven approach using the HTK tool. Different number of Gaussian mixtures and different number of HMM states are investigated in order to reach the best configuration model for each phoneme.

3. Related Work

Phoneme recognition and transcription in continuous speech recognition has been addressed by many researchers. Lee and Hon [8] used a large network of trained phonemic HMMs where the maximum likelihood state sequence of the network gave the recognized phoneme sequence. A multi-state fixed structure with three observation probability represented a phoneme. The acoustic variability was better characterized, even though the phonemes duration and structure were not modeled accurately. Levinson *et al.* [9] used a single large Continuous Variable Duration Hidden Markov Model (CVDHMM) with a number of states equal to the number of phonemes in the vocabulary. A single CVDHMM state is characterizing the whole phoneme. The state models both, the acoustic variability and the durational variability of each phoneme. Levinson *et al.* [9] reported a reasonable performance because of the external knowledge that was incorporated into the model, such as the phoneme duration and the bigram probabilities. Elmahdy *et al.* [5] proposed an approach for rapid phonetic transcription of dialectal Arabic, where the Arabic Chat Alphabet (ACA) was used. The proposed approach represented an alternative way to conventional phonetic transcription methods. Results showed that the ACA based approach outperforms the graphemic baseline while it performs as accurate as the phoneme recognition based baseline with a slight increase in WER. Bayeh *et al.* [4] presented a study of manual and data driven association of words and phonemes. Their study was based on defining the speech recognition system in a target language based on acoustic models from another source language. They found that phoneme to phoneme association was more practical but, words transcription provided better results. Experiments were conducted with French as the source language and Arabic as the target language.

Alghamdi *et al.* [3] proposed a new labeling scheme which was able to cover all the quranic sounds and their phonological variations. They presented a set of labels that covers all Arabic phonemes and their allophones. They showed how it can be efficiently used to segment the quranic corpus. The authors claimed that the initial results were encouraging.

4. Methodology and Experimental Setup

The methodology we followed to transcribe Arabic phonemes in a data driven approach is shown in Figure 1. The methodology involves the following steps:

1. Data preparation, which includes:
    a. Defining a proper finite state recognition grammar.
    b. Building the HMM models (Prototypes) for each Phoneme.
2. Training the HMM models.
3. Phoneme recognition and transcription.
4. Experimentation to find the best HMM configuration.
5. Adding the language model.

Details of Tasks 1-3 are given in the remainder of this section. Section 5 explains tasks 4 and 5.

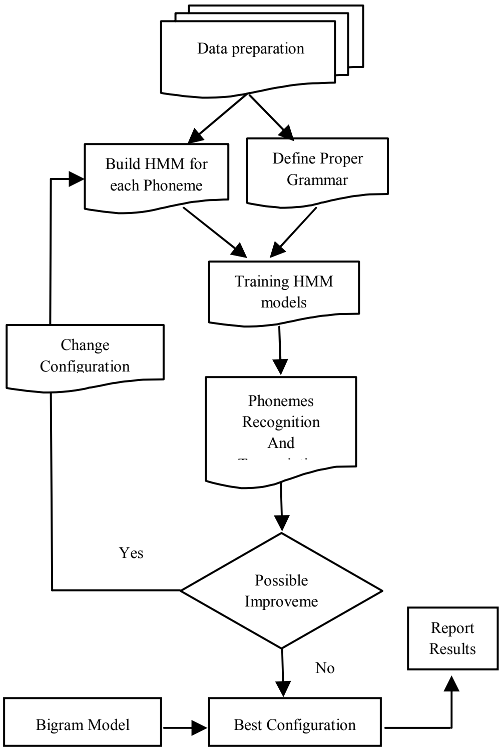


Figure 1. Arabic phoneme transcription methodology.

4.1. Data Preparation

Five hour recordings of modern Arabic speech from different TV-News clips have been developed and transcribed by King Fahd University of Petroleum and Minerals (KFUPM) with the support of King Abdul-Aziz City for Science and Technology (KACST). This corpus is considered as one of the few available resources to modern standard ASR systems.

The corpus consists of 10-20 millisecond utterances with 16 KHz sampling rate, along with their corresponding MFCC files and their transcriptions. In addition, it contains a list of Arabic phonemes, an Arabic dictionary and other script files used for manipulating corpus information. This corpus was used in the development of a continuous ASR system [1, 2].

Using CMU sphinx, the trigram word transcription and time alignment of all utterances of the corpus was produced. Table 1 shows the transcription of the word "إرتفعت" (EIHRTAEFAEAIAET) as an example of, trigram word transcription done by CMU sphinx. Providing the CMU Sphinx with the MFCC feature files of the training set corpus and its corresponding transcription, a mechanism of forced alignment was done by Sphinx to generate the phoneme and word segmentation for all utterances of the training set. The testing set of utterances used in phoneme transcription was not seen before for the purpose of evaluation.

CMU Sphinx produces a start and end frame numbers that are associated with each phoneme utterance, as shown in Table 1. The HTK, on the other hand, uses a start and end frame numbers based on integer multiples of 100ns, as shown in Table 2. Since the output of the CMU Sphinx is fed as input to the HTK tool, we had to transform the CMU sphinx frame labeling into its equivalent HTK tool labeling and remove all other columns. Table 2, shows the HTK tool labelingof the word "إرتفعت".

Table 1. The word "إرتفعت" on the (Left) part, trigram word phoneme segmentation on (Right) part.

CMU Sphinx Labeling							
Word Segmentation			Phoneme Segmentation				
StaFrm	EdFrm	Word	StaFrm	EdFrm	Phone	Prev	Next
0	12	<s>	0	12	SIL		
13	15	<s>	13	15	SIL		
16-59		إرتفعت	16	21	E	SIL	IH
			22	24	IH	E	R
			25	27	R	IH	T
			28	30	T	R	AE
			31	33	AE	T	F
			34	36	F	AE	AE
			37	39	AE	F	AI
			40	44	AI	AE	AE
			45	51	AE	AI	T
			52	59	T	AE	E

Table 2. The corresponding HTK labeling of the word in Table 1.

HTK Labeling		
StartFrm	EndFrm	Phone
00000	1200000	SIL
1300000	1500000	SIL
1600000	2100000	E
2200000	2400000	IH
2500000	2700000	R
2800000	3000000	T
3100000	3300000	AE
3400000	3600000	F
3700000	3900000	AE
4000000	4400000	AI
4500000	5100000	AE
5200000	5900000	T
3700000	3900000	E

Another issue to handle was the incompatibility of the MFCC feature files included in the corpus and the MFCC feature files that are accepted by the HTK tool. Therefore, we had to regenerate the MFCC files from the WAV files of the corpus using the HCopy command in HTK. It is worth mentioning that the HCopy command requires proper configuration parameters according to the input WAV files. Table 3 shows the configuration parameters that were most suitable to the corpus WAV files.

Table 3. Configuration parameters for HCopy command

HTK Tool - HCopy Configuration Parameters	
Variable	Value
Targetkind	MFCC_0_D_A
Targetrate	100000.0
Savecompressed	F
Savewithcrc	F
Windowsize	250000.0
Usehamming	F
Preemcoef	0.97
Numchans	26
Ceplifter	22
Numceps	12
Enormalize	F

HTK supports many different types of MFCC feature vectors. The two most-commonly used types in speech recognition are the basic type, MFCC_0 with 13 features frame length, and the extended one, MFCC_0_D_A with 39 features frame length. Since, CMU Sphinx uses the extended one, we selected the MFCC_0_D_A type[1].

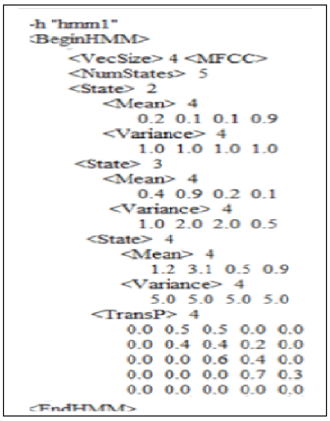
Finally, the corpus was divided into two parts; one part for training and the other part for testing. The size of training part was 70% of the corpus. The remaining 30% of the corpus was used for testing.

4.2. Building HMM Models

We created an HMM definition file (Prototype) for each Arabic phoneme. Initially, we used 3-emitting state HMM with one Gaussian mixture for all the phonemes. The 3-emitting state HMM is represented by 5-states in the HTK tool (one entry state, 3-emitting states and one final state). The feature vector used was MFCC_0 with13 feature values for each frame. Figure 2 shows a simple HMM prototype file of 3-emitting states. The function of the prototype is to describe the form and topology of the HMM. The initial numbers representing transition sate probabilities used in the definition are not important since they will be updated by the training process.

Different numbers of Gaussian mixtures were used along with different numbers of HMM states in modeling Arabic phonemes. We will first evaluate the accuracy without using any language model. Then, we will add the language model to the configuration of the highest accuracy.

[1] The '0' character means 13 features per frame, 'D' character stands for Delta (first difference) and A stands for Acceleration (Second difference).



```
-h "hmm1"
<BeginHMM>
      <VecSize> 4 <MFCC>
      <NumStates>  5
      <State>  2
          <Mean>  4
              0.2  0.1  0.1  0.9
          <Variance>  4
              1.0  1.0  1.0  1.0
       <State>  3
          <Mean>  4
              0.4  0.9  0.2  0.1
           <Variance>  4
              1.0  2.0  2.0  0.5
         <State>  4
            <Mean>  4
               1.2  3.1  0.5  0.9
            <Variance>  4
               5.0  5.0  5.0  5.0
          <TransP>  4
             0.0  0.5  0.5  0.0  0.0
             0.0  0.4  0.4  0.2  0.0
             0.0  0.0  0.6  0.4  0.0
             0.0  0.0  0.0  0.7  0.3
             0.0  0.0  0.0  0.0  0.0
<EndHMM>
```

Figure 2. Simple HMM prototype [12].

Different numbers of Gaussian mixtures were used along with different numbers of HMM states in modeling Arabic phonemes. We will first evaluate the accuracy without using any language model. Then, we will add the language model to the configuration of the highest accuracy.

4.3. Defining Proper Grammar, Dictionary, and Network Lattices

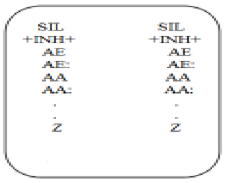
Using HTK grammar definition language, we define the grammar for each Arabic utterance as shown in Figure 3.

```
PHONEME=AE|AE: | AA| AA: | … | Z
(SIL SAL<$PHONEME> SIL SIL)
```

Figure 3. Recognition grammar.

The first part of the grammar is the variable PHONEME which defines all possible Arabic phones including Silence, (SIL) and Inhalation (+INH+). The character '|' represents the logical OR operator. The second part represents the utterance where two silences are added, at the beginning and at the end of each utterance. At the middle is the reference for the variable PHONEME inside the characters '<>' which represent the repetition.

Using the HTK dictionary definition rules, we define the phonemes' dictionary. It consists of two identical columns containing the list of Arabic phonemes based on the HTK tool instructions for recognition at the phoneme level [12]. Figure 4 shows part of the dictionary.



```
  SIL        SIL
+INH+      +INH+
  AE         AE
  AE:        AE:
  AA         AA
  AA:        AA:
   .          .
   .          .
   Z          Z
```

Figure 4. Part of the dictionary.

The dictionary could be generated automatically through the HTK. It could also be prepared manually using any text editor [12].

The phoneme lattice network is generated after setting the proper grammar and the dictionary. The Lattices network is generated by providing the dictionary and the grammar as input to the HTK-HParse and the HTK-HGen commands. HTK-HGen command is then used to confirm the correctness of the grammar by automatically generating set of predefined number of utterances.

4.4. Training HMM's

At this point and after the configuration was set and the grammar was defined, one can start the HMM training process. The HMM models of the phonemes are first initialized using the Baum Welch algorithm. The HTK tool supports two types of initialization. The first type is applied when the boundaries of phonemes are known. The second one is applied when the boundaries of phonemes are not known, which is called flat initialization. Since, the phoneme boundaries of our data are known, the first type of initialization is applied using the HInit command of HTK.

Figure 5, shows the needed inputs and the resulting outputs of the initialization process, for more details on HInit-HTK command see [12]. After initializing all the HMMs, the first iteration of the iterative Baum Welch algorithm is executed using the HTK-HRest command.

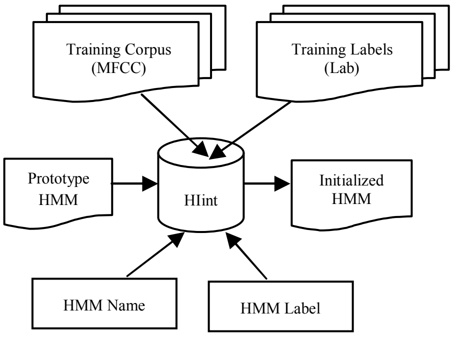


Figure 5. Initialization from prototype [6].

HTK-HRest command functionality is very similar to the HTK-HInit command, except that it expects that the input HMM definition has already been initialized. It uses the Baum-Welch re-estimation algorithm in place of the Viterbi training [12]. All initialized HMMs are put in one file called the Master Macro File (MMF).

Next, we carry out our phoneme recognition and transcription task shown in Figure 1, without using any language model in order to determine the most accurate configuration achievable. Figure 6 shows details of this process. The training module receives the training MFCC files containing the features of the segmented speech, its corresponding text, and the initialized HMM for each phoneme. The training module produces trained HMM's. The trained HMMs and the defined grammar are used to recognize the MFCC files extracted from the testing set utterances, producing the resulting transcription.

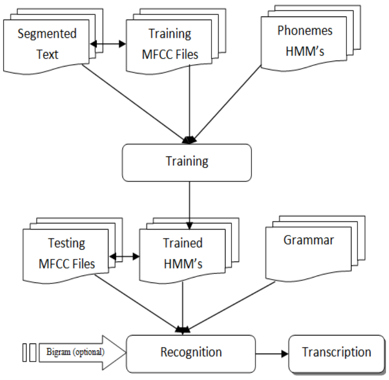


Figure 6. Phonemes recognition and transcription process.

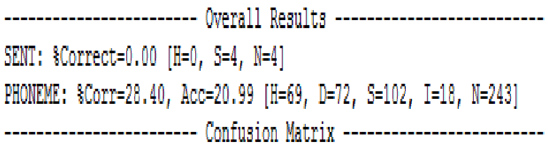
This process is repeated for different HMM configurations until no significant improvement is achieved. At this point, the HTK-H build command, used to build the language model, adds the bigram probability to the phonemes network. Details of these experimentations are shown in section 5.

5. Experimental Results

A set of experiments were conducted. Each experiment has its own configuration based on the study and analysis of the results of the preceding experiments. A total of 25 experiments were conducted in order to find the best configuration. Below, we report the results of four of such experiments that reported the highest significant improvements based on our sensitivity analysis. This was done manually by investigating the confusion matrix of each experiment.

5.1. Experiment 1: Basic Feature Vector with 3 HMM-Emitting States

In this experiment, we used 3 HMM-emitting states for all phonemes, and extracted the 13-value basic features from the WAV files. No Gaussian mixtures were used. The percentage of correctly recognized phonemes was 28.4%. A portion of the output file generated by the HTK-H result command for aligning the labeled tested utterances with the reference labeling is shown in Figure 7. This low recognition rate was due to the usage of the basic features which were not an adequate set of features that can suitably represent wav files for different phonemes.



```
----------------------- Overall Results -------------------------
SENT: %Correct=0.00 [H=0, S=4, N=4]
PHONEME: %Corr=28.40, Acc=20.99 [H=69, D=72, S=102, I=18, N=243]
----------------------- Confusion Matrix ------------------------
```

Figure 7. First experiment results.

Once, the HTK-H result is executed, then, insertion errors *I*, substitution errors *S*, deletion errors *D* and the total number of labels in the reference transcription *N* are calculated and reported as shown above. The H-value in this output refers to the number of correct hits, i.e., the number of correctly recognized phonemes. The percentages of correctness (denoted by % Corr) and accuracy (denoted by % Acc) are based on the values of *I*, *S*, *D* and *N*. Those percentages are computed as follows [12]:

$$\textit{Percent Correct} = \frac{N - D - S}{N} \times 100\% \qquad (1)$$

$$\textit{Percent Correct} = \frac{N - D - S - I}{N} \times 100\% \qquad (2)$$

5.2. Experiment 2: Extended Feature Vector with 3 HMM-Emitting States 1

In the second experiment, we used the extended feature vector instead of the basic one with 4-Gussaian mixtures. The recognition rate increased to 43.3%. We conclude that this significant increase in accuracy is attributed to our use of the extended feature vector of 39 features to represent the wave files along with the use of the 4-Gussaians. Therefore, all subsequent experiments were conducted on the extended feature vector.

5.3. Experiment 3: Extended Feature Vector with Varying Number of HMM-States 1

In the third experiment, we used the extended feature vector with 4-Gussian mixture. In addition, we used different number of emitting states as follows: The phonemes representing long and intermediate vowels such as /AA: /, /AE: /, /AY/, /UW/, /AW/ and /AH:/ were assigned 4-HMM emitting states. The un-voiced stop phonemes /E/, /T/, /K/ and /Q/ were assigned 1-HMM emitting state, whereas the voiced-stop phonemes /B/, /D/ and /DD/ were assigned 2-HMM emitting states. The remaining phonemes were assigned 3 HMM emitting states. Table 4 shows the complete list of the Arabic phonemes with their corresponding Arabic letters including some examples of using Arabic vowels allophones.

Table 4. Arabic phoneme list.

Phoneme	Arabic Alphabet	Example	Phoneme	Arabic Letter
/AE/	ـَ	بَ	/Z/	ز
/AE: /	ـَا	بَاب	/S/	س
/AA/	ـَ	خَ	/SH/	ش
/AA: /	ـَ	خَاب	/SS/	ص
/AH/	ـَ	قَ	/DD/	ض
/AH: /	ـَ	قَال	/TT/	ط
/UH/	ـُ	بُ	/DH2/	ظ
/UW/	ـُو	دُون	/AI/	ع
/UX/	ـُ	غُصن	/GH/	غ
/IH/	ـِ	بِنت	/F/	ف
/IY/	ـِي	فِيل	/V/	-
/IX/	ـِ	صِنف	/Q/	ق
/AW/	ـَو	لَوم	/K/	ك
/AY/	ـَي	صَيف	/L/	ل
/E/	ء	دفـء	/M/	م
/B/	ب		/N/	ن
/T/	ت		/H/	هـ
/TH/	ث		/W/	و
/JH/	ج		/Y/	ي
/G/	ج			
/ZH/	ج			
/HH/	ح			
/KH/	خ			
/D/	د			
/DH/	ذ			
/R/	ر			

The percentage of correctly recognized phonemes in this experiment increased to 51.3%. This shows that using different number of HMM emitting states had a positive impact on the correctness.

5.4. Experiment 4: Extended Feature Vector with Varying Number of HMM-States 2

Based on the confusion matrix of Experiment 3, it was evident that the phoneme /T/ requires no more than one emitting state. In addition, the number of emitting states for Phonemes /E/, /K/ and /Q/ were changed to 3, instead of 1. Similarly, Phonemes /B/, /D/ and /DD/ were changed to 3, instead of 2 emitting states. The number for Phoneme /AA:/ was reduced from 4 to 3 emitting states. Finally, Phonemes /UW/ and /AE/ were assigned 4-HMM emitting states. With these, the percentage of correctness increased to 56.79 %.

The results of Experiments 3 and 4 highlighted the importance of carrying out a detailed analysis of the various forms a certain phoneme occurs in the Arabic speech corpus, in order to systematically determine the most suitable number of HMM emitting states for each phoneme. The next section summarizes our findings with respect to this issue. For more details, refer to [10].

5.5. Statistical Analysis

Determining the most suitable number of HMM states for each phoneme depends on the various statistical properties of this phoneme that can be derived from its corresponding sample occurrences in the corpus. Hence, we carried out a statistical investigation of various phoneme features, including phoneme lengths frequency, phoneme lengths probability, phoneme frequency, phoneme lengths mode (the most frequent value within a set of values), phoneme lengths median, phoneme bigram probability, phoneme trigram frequency and the occurrence probability of the phonemes in the corpus utterances. The results of this study are shown below.

Figure 8 presents the median length, in frames, of all Arabic phoneme lengths after sorting them in non-decreasing order. However, due to space limitations, selected phonemes are identified on the graph's *x*-axis. The full graph information for this figure and subsequent figures can be found in [11]. One can see that Phoneme /E/ is considered to be a cluster on its own, Phonemes /AE/, /UH/, /R/ and /N/ belong to the same cluster and so on.

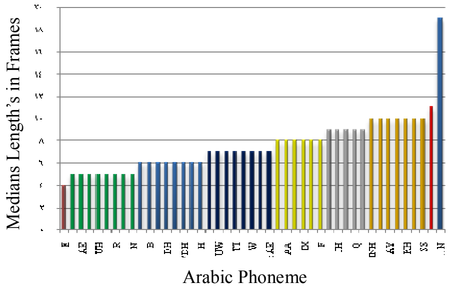


Figure 8. Sorted list of Arabic phoneme length's medians.

Figure 9 displays the most frequent length that each phoneme occurred with, in all utterances.

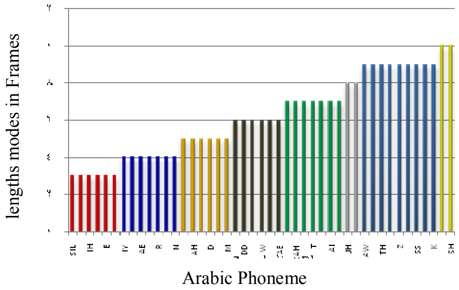
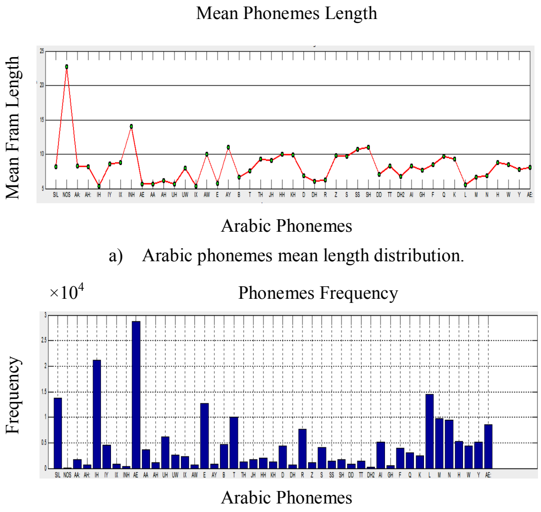


Figure 9. Sample of Arabic phonemes lengths modes after sorting.

In addition, Figure 10 shows additional information related to the phonemes mean length distribution and their frequency.



a) Arabic phonemes mean length distribution.

b) Arabic phonemes frequency distribution.

Figure 10. Arabic phonemes mean length and frequency distribution.

Table 5 gives details of certain phoneme lengths probabilities extracted from the corpus. This table highlights the maximum probability phoneme length may occur.

Table 5. Sample of phonemes length probability.

Prob of Length IS	AA:	AH:	IH	IY	IX:	AE	AA	AH
3	0.0295	0.0225	0.2286	0.1032	0.0522	0.1496	0.1534	0.0557
4	0.0416	0.0435	0.2271	0.1186	0.0842	0.2410	0.2188	0.1515
5	0.0960	0.0810	0.1904	0.0925	0.1091	0.2235	0.2432	0.2418
6	0.1393	0.0945	0.1414	0.0967	0.1210	0.1548	0.1606	0.2391
7	0.1578	0.1814	0.0861	0.1086	0.1293	0.0850	0.0879	0.1305
8	0.1410	0.1694	0.0492	0.1082	0.1186	0.0491	0.0466	0.0657
9	0.1052	0.1439	0.0267	0.0862	0.0913	0.0251	0.0208	0.0383
10	0.0879	0.1109	0.0157	0.0652	0.0652	0.0165	0.0161	0.0173
11	0.0607	0.0540	0.0092	0.0450	0.0558	0.0109	0.0086	0.0173
12	0.0555	0.0315	0.0060	0.0363	0.0474	0.0092	0.0078	0.0055
13	0.0150	0.0255	0.0047	0.0215	0.0273	0.0075	0.0075	0.0119
14	0.0260	0.0150	0.0029	0.0224	0.0154	0.0059	0.0067	0.0036
15	0.0133	0.0045	0.0019	0.0167	0.0107	0.0055	0.0033	0.0100

Figure 11 shows the shifted distribution of the length probability between long vowel phoneme and the short version of it.

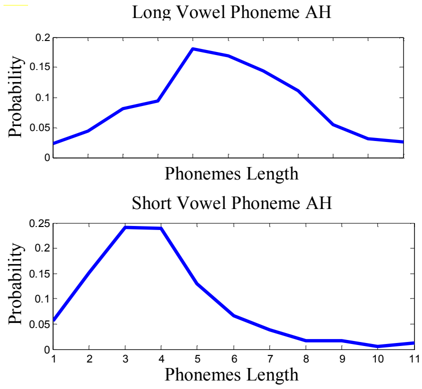


Figure 11. Phoneme length probability distribution.

Another important table that was used to enhance the recognition process is the bigram language model table. Table 6 shows the bigram for the phonemes AE and AH as a sample.

Table 6. AE and AH phonemes bigram table.

AE Phoneme (P (Ph| AE))			AH Phoneme (P (Ph| AH))		
Prob	**Phoneme**	**Next Phone**	**Prob**	**Phoneme**	**Next Phone**
-3.127	AE	+INH+	-2.8637	AH	AE
-3.9154	AE	+NOISE+	-3.3408	AH	AE:
-2.8972	AE	AE	-1.7498	AH	AI
-4.2834	AE	AE:	-1.3452	AH	B
-1.2953	AE	AI	-1.4324	AH	D
-1.4707	AE	B	-3.3408	AH	DH
-1.4636	AE	D	-1.5279	AH	E
-2.1728	AE	DD	-1.2581	AH	F
-1.8058	AE	DH	-2.4957	AH	GH
-2.3678	AE	DH2	-1.0553	AH	H
-1.1772	AE	E	-1.6166	AH	HH
-1.6591	AE	F	-2.4957	AH	IH
-2.2791	AE	GH	-3.3408	AH	JH
-0.9939	AE	H	-1.9095	AH	KH
-1.6503	AE	HH	-0.9481	AH	L
-2.6777	AE	IH	-1.0553	AH	M
-4.2834	AE	IY	-1.0737	AH	N
-1.7972	AE	JH	-1.57	AH	Q
-1.6927	AE	K	-0.8081	AH	R
-1.8623	AE	KH	-2.1104	AH	S
-1.1312	AE	L	-2.4957	AH	SH
-1.1527	AE	M	-1.9095	AH	SS
-1.1432	AE	N	-0.9176	AH	T
-1.5257	AE	Q	-2.0186	AH	TT
-1.1464	AE	R	-1.7968	AH	W
-1.4783	AE	S	-1.4896	AH	Y

The bigram probability appearing in Table 6 is given using log base 10, as generated by the HTK-tool.

5.6. Experiment with Bigram-Language Model

Based on the results of the previous section, it is evident that phonemes have been clustered in different statistical features in such a way that one can choose a suitable number of HMM emitting states based on that clustering.

It is also clear that using the bigram language model can be helpful in achieving better recognition rates. With respect to the number of HMM states, and based on our experimentation, we chose the phoneme median length results to classify the phonemes based on the assigned number of HMM states.

Each cluster in Figure 8 corresponds to a numbers of HMM states. One exception to this was the Phoneme /T/, in which one HMM emitting state was assigned to it based on our previous experimental work. Table 7 shows the various clusters of phonemes and the assigned number of HMM emitting states for each one of them.

Table 7. Arabic phonemes clusters.

Phonemes	**Arabic Letter**	**Number of HMM States**
Class 1		
/E/	ء	2
Class 2		
/IH/	ـِ	3
/AE/	ـَ	3
/AA/	ـَ	3
/UH/	ـُ	3
/IX/	ـِ	3
/R/	ر	3
/L/	ل	3
/N/	ن	3
Class 3		
/AH/	ـَ	4
/B/	ب	4
/D/	د	4
/DH/	ذ	4
/DD/	ض	4
/DH2/	ظ	4
/M/	م	4
/H/	هـ	4
Class 4		
/IY/	ـِي	4
/UW/	ـُو	4
/T/	ت	1
/TT/	ط	4
/GH/	غ	4
/W/	و	4
/Y/	ي	4
/AE:/	ـَا	4
Class 5		
/AA:/	ـَ	5
/AH:/	ـَ	5
/UX/	ـُ	5
/AI/	ع	5
/F/	ف	5
Class 6		
/TH/	ث	5
/JH/	ج	5
/Z/	ز	5
/Q/	ق	5
/K/	ك	5
Class 7		
/AW/	ـَو	6
/AY/	ـَي	6
/HH/	ح	6
/KH/	خ	6
/S/	س	6
/SH/	ش	6
/SS/	ص	6

In our experimentation, we used 8-Gussian mixtures except for the /SIL/ model, in which 4-Gussian mixtures was used. This experiment achieved a phoneme correctness percentage of 96.3%, which is a significant improvement over all our previous experiments that did not include the language model. In addition, these results are very encouraging, compared to other experiments conducted for the English language, where a percentage correctness of 83% was reported in [7].

5.7. Experiments Summary

Table 8 summarizes the results of the experiments that have been conducted.

Table 8. Experiment's summary.

Exp No	HMMStates	Feature Vector	Vector Length	Language Model	No. Gaussian Mixtures	Accuracy
1	3	MFCC-0	13	N/A	1-GM	28.4%
2	3	MFCC-0-D-A	39	N/A	4-GM	43.3%
3	Different (1,2,3,4)	MFCC-0-D-A	39	N/A	4-GM	51.3%
4	Changed Based On Exp. 3	MFCC-0-D-A	39	N/A	4-GM	56.79%
5	Changed Based On Statistical Study	MFCC-0-D-A	39	Bigram	4-GM for /SIL/ 8-GM for others	96.3%

6. Conclusions and Future Work

In this paper, we used a data driven approach to recognize and transcribe the Arabic phonemes. We presented the process of phoneme recognition using HMM, implemented with the HTK toolkit. Five experiments were conducted using different HMM parameters. These parameters were mainly set after classifying the phonemes into clusters based on our statistical study for the phonemes length, frequency, median, mode, etc., The maximum recognition accuracy achieved was 96.3% after employing the bigram language model. A grammar, Phoneme network lattices and a dictionary were generated for the purpose of Arabic phoneme transcription.

The maximum number of Gaussian mixture used was 8. This number was limited by the size of the existing corpus. It is worth mentioning that more elaborate Arabic speech corpora are still under development with KACST support, which may further improve the recognition rate upon using a higher number of Gaussian mixtures. In addition, we look forward to studying the viability of our proposed methodology on out of vocabulary word recognition, and subsequently on non-MSA (dialect speech) recognition in the near future.

Based on the results of this research and when combining the phoneme recognition with the word language model, we expect a significant enhancement in the continuous ASR.

References

[1] Abushariah M., Ainon R., Zainuddin R., ElshafeiM., and Khalifa O., "Arabic Speaker-Independent Continuous Automatic Speech Recognition based on a Phonetically Rich and Balanced Speech Corpus," *the International Arab Journal of Information Technology*, vol. 9, no.1, pp. 84-93, 2012.

[2] Abuzeina D., Al-Khatib G., Elshafei M., and Al-Muhtaseb H., "Within-Word Pronunciation Variation Modeling for Arabic ASRs: A Direct Data-Driven Approach," *International Journal of Speech Technology*, vol. 15, no. 2, pp. 65-75, 2012.

[3] Alghamdi M., Elshafei M., and Al-Muhtaseb H., "Arabic Broadcast News Transcription System," *International Journal of Speech Technology*, vol. 10, no. 4, pp. 183-195, 2007.

[4] Bayeh R., Lin S., Chollet G., and Mokbel C., "Towards Multilingual Speech Recognition using Data Driven Source/Target Acoustical Units Association," *in Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, Montreal, Canada, pp. 521-524, 2004.

[5] Elmahdy M., Gruhn R., Abdennadher S., and Minker W., "Rapid Phonetic Transcription using Everyday Life Natural Chat Alphabet Orthography for Dialectal Arabic Speech Recognition," *in Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, Prague, Czech Republic, pp. 4936-4939, 2011.

[6] http*://www.info2.uqam.ca/~boukadoum_m/DIC9315/Notes/Markov/HTK_basic_tutorial.pdf.*

[7] Kim Y., Chan Y., Evermann G., Gales F., Mrva D., Sim C., and Woodland C., "Development of the CU-HTK 2004 Broadcast News Transcription Systems," *in Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, Philadelphia, USA, pp. 861-864, 2005.

[8] Lee F. and Hon W., "Speaker independent Phone Recognition using Hidden Markov Models," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 37, no. 11, pp. 1641-1648, 1989.

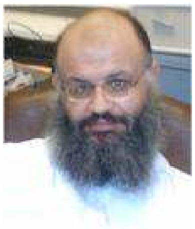
[9] Levinson E., Liberman Y., Ljolje A., and Miller G., "Speaker Independent Phonetic Transcription of Fluent Speech for Large Vocabulary Speech Recognition," *in Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, Glasgow, Scotland, pp 441-444, 1989.

[10] Liang S., Lyu Y., and Chiang C., "Using Speech Recognition Technique for Constructing a Phonetically Transcribed Taiwanese (Min-Nan) Text Corpus," *in Proceedings of The 9th International Conference on Spoken Language Processing*, Pittsburgh, USA, pp. 193-196, 2006.

[11] Nahar K., Elshafei M., Al-Khatib G., Al-Muhtaseb H., and Alghamdi M., "Statistical Analysis of Arabic Phonemes for Continuous Arabic Speech Recognition," *International Journal of Computer and Information Technology*, vol. 1, no. 2, pp. 49-61, 2012.

[12] Young J., Evermann G., Gales F., Hain T., Kershaw D., Moore G., Odell J., Ollason D., Povey D., Valtchev V., and Woodland C., The HTK Book (Version 3.4)., available at: http://speech.ee.ntu.edu.tw/homework/DSP_HW2-1/htkbook.pdf, last visited 2006.

Khalid Nahar assistant professor Computer Science Department-Tabuk University, KSA (Unpaid Leave) From Yarmouk University, Jordan.He is an assistant professor at Tabuk University, Saudi Arabia. He received his BS and MS degrees in computer science from Yarmouk University-Jordan in 1992 and 2005 respectively. He received his PhD in Computer Science and Engineering from King Fahd University of Petroleum and Minerals (KFUPM). He worked at Yarmouk University as Teacher Research Assistant for 7 years and 5 years as a lecturer, for now he is working at Tabuk University-KSA. His research interests include continuous speech recognition, arabic computing, natural language processing, multimedia computing, content-based retrieval, artificial intelligence, and software engineering. He participated in funded research projects.



Husni Al-Muhtaseb Assistant Professor, Computer Science Department Husni Al-Muhtaseb Obtained a PhD degree from the Department of Electronic Imaging and Media Communications (EIMC) of the School of Informatics in the University of Bradford, UK in 2010. He received his M.S. degree in computer science and engineering from King Fahd University of Petroleum and Minerals (KFUPM), Dhahran, Saudi Arabia, in 1988 and the B.E. degree in electrical engineering, computer option, from Yarmouk University, Irbid, Jordan in 1984. He is currently an Assistant Professor of Information and Computer Science at KFUPM. He was working as an Instructor with the same department from 1992 to 2010. He worked as a technical consultant with the dean of admissions and registration from 1996 to 2007. From 1988 to 1992, he worked as lecturer at KFUPM. From 1984 to 1988, he worked as Research and Teaching Assistant at Yarmouk University and KFUPM. His research interests include software development, Arabic Computing, computer Arabization, Arabic OCR, e-learning & online tutoring and natural Arabic understanding. he developed the first course in the world on Arabization of Computers. Currently, several Universities and colleges are adapting the course. He has participated in several industrial projects with different institutes/ organizations including, KACST, STC, MOHE and Aramco. He also worked as a consultant for different entities including KFUPM schools and Ministry of education. He has more than 60 research publications. He got the first excellence award in instructional Technologies at KFUPM for year 2007.

Wasfi Al-Khatib Assistant Professor King Fahd University of Petroleum and Minerals, Saudi Arabia. He received his BS degree in computer science from Kuwait University in 1990, and his MS degree in computer science and PhD in Electrical and Computer Engineering from Purdue University in 1995 and 2001, respectively. He worked at Wright State University in Dayton, Ohio as an assistant professor from 2001-2002. His research interests include Arabic computing, multimedia computing, content-based retrieval, artificial intelligence, and software engineering. He participated in funded research projects and supervised many graduated students. He also participated in curriculum development and ABET assessment accreditation efforts, both at the department level and at the University level. He is a member of the ACM and the IEEE Computer Society.

Moustafa Elshafei Professor, Systems Engineering Department he received his PhD (with Dean List) from McGill University, Canada, in electrical engineering in 1982. Since then, he has accumulated over 31 years of both academic and industrial experience. He is sole inventor/co-inventor of 13 US and international patents. He has over 150 publications in international journals, conferences, and technical reports. He was the PI/CI of many government funded projects exceeding 8 million SR, and he was also involved in many internally funded or industry funded projects. His research interest includes Arabic speech processing, digital signal processing, and intelligent instrumentation.

Mansour Alghamdi, Professor in Phonetics, King Abdul-Aziz City for Science and Technology, Saudi Arabia. he gained a PhD in phonetics at Reading University, UK, in 1990. He has more than 80 published books, journal papers and conference presentations, has 3 patents, participated in more than 20 scientific projects at KACST, KFUPM and KSU as the principle investigator or co-investigator, lectured in several institutions on the applied phonetics including: computational linguistics, speech therapy, translation, first language acquisition and foreign language learning. He is now the director of General Directorate of Scientific Awareness and Publishing and the director of the National Program for Digital Content at King Abdul-Aziz City for Science and Technology, Riyadh.