CERTIFICATE
OF PARTICIPATION

Paper Title: Arabic Speech Synthesis System based on HMM (E14)

This is to certify

Aissa Amrouche, Scientific and Technical Research Center for the Development of the Arabic Language, Algeria

has attended and delivered an oral presentation on 2019 6th International Conference on Electrical and Electronics Engineering (ICEEE 2019) held during April 16-17, 2019 in Istanbul, Turkey, which is supported by Gazi University and co-supported by Batman University.

ICEEE 2019
Conference Committee

2019 6th International Conference on Electrical and Electronics Engineering (ICEEE)

Arabic Speech Synthesis System Based on HMM

Aissa Amrouche
Laboratory of Spoken communication and signal processing, USTHB
Scientific and Technical Research Centre for The Development of Arabic Language
CRSTDLA, Algiers, Algeria
e-mail: amrouche_a@yahoo.fr

Ahcène Abed
Signal and Communication Laboratory, USDB
Blida, Algeria
e-mail: abedahcene@gmail.com

Leila Falek
Laboratory of Spoken Communication and signal processing, USTHB
Algiers, Algeria
lfalek@hotmail.fr

***Abstract*—The work presented in this paper is about Text-to-Speech (TTS) synthesis and, more particularly, about statistical speech synthesis using the Hidden Markov Models (HMM). The main objective of this work is to study the functioning of the HMM-based speech synthesis system (HTS) and the implementation of this method to create a system that produces understandable speech output for a given Arabic text. We have done a brief description of the statistical parametric speech synthesis based on HMM, the steps followed to implement this method for Arabic language. Finally, for the evaluations of the system are based on subjective mean opinion score and objective tests. Regarding the intelligibility, naturalness aspects (listening) and the quality (Perceptual Evaluation of Speech Quality (PESQ)).**

Keywords-text-to-speech synthesis; acoustic vectors; minimum distance; likelihood; hidden Markov models (HMM); HTS

I. INTRODUCTION

Since the mid-2000s, two methods are predominant in the field of speech synthesis: unit selection and parametric synthesis based on Hidden Markov Models (HMM) [1]. Special attention is being paid to the HTS, which is the best-known and most widely used representative for HMM synthesis. Indeed, this system has the advantage of being able to produce an intelligible and fluid synthesis using less data than a unit selection system [2]. These speech synthesis systems mainly intend for embedded applications in telecommunication systems such as smartphones or other wireless devices, which today take a very important place in our lives. On the other hand, contrary to synthesis by selection, the synthesis signal produced by HTS is low quality.

To perform a synthesis using the HTS system, it is necessary to associate to the signal a description based on a consistent set of linguistic and prosodic descriptors [3]. We can distinguish two important limitations resulting from the combinatorics resulting from this description. First, defining a corpus to cover all possible combinations is impractical. This implies that, during the synthesis phase, combinations, which it is desired to synthesize, may not have been seen during the learning phase. In addition, the HTS system is based on statistical modeling. Thus, the number of occurrences associated with a combination of descriptors, in the learning corpus, may also be too small to obtain a relevant model. Although the HTS system proposes the use of decision trees to overcome these problems, which is much more critical, the advantages of the HTS system is that can use a small size-learning corpus to perform the speech synthesis.

The aim of our work is to contribute to the implementation of a system that can produce natural speech sounding and produce an output that resembles to acoustic and prosodic characteristics of the original speaker. For this purpose, we have study the principle of the HMM-based speech synthesis system (HTS). Indeed, the implementation of a synthesis system is a very difficult task, which requires the realization of several modules such as database, phonetic orthographic transcription, automatic selection of units in the database, to obtain a synthetic speech signal more or less faithful to the input text.

II. HMM SPEECH SYNTHESIS SYSTEM

In the HMM-based speech synthesis [4], also called statistical parametric synthesis, speech waveforms are generated using trained context-dependent HMMs which models different speech parameters such as spectrum, excitation, and phoneme duration. A well-known example of HMM-based speech synthesizer is HMM-based Speech Synthesis System (HTS) [3]. The system as shown in Figure 1 consists of two stages:

978-1-7281-3910-4/19/$31.00 ©2019 IEEE
DOI 10.1109/ICEEE2019.2019.00022

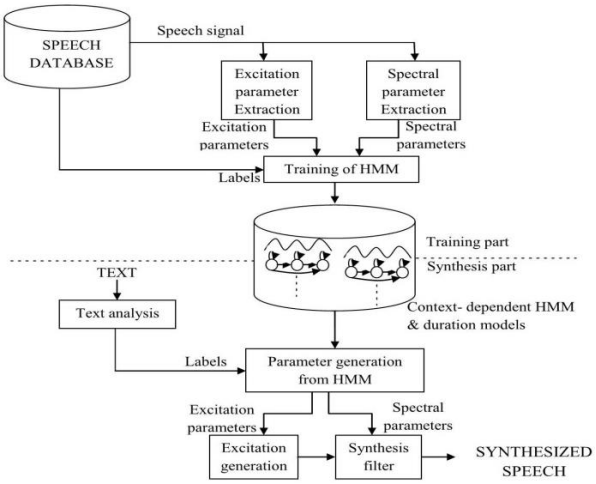

Figure 1. Overview of HTS system [3].

- The training stages;
- The synthesis stages.

Heiga Zen, and et al, illustrated the stages of HTS in [3] as follows:

The training part is similar to that used in speech recognition systems. The main difference is that both spectrum (mel-cepstral coefficients, and their dynamic features) and excitation (logarithmic fundamental frequencies (Log F0) and its dynamic features) parameters are extracted from a speech database and modeled by context-dependent HMMs (phonetic, linguistic, and prosodic contexts are taken into account). To model variable dimensional parameter sequence such as (Log F0) with unvoiced regions properly, multispace probability distributions (MSD) are used. Each HMM has state duration probability density functions (PDFs) to capture the temporal structure of speech. As a result, the system Models spectrum, excitation, and durations in a unified HMM framework. The synthesis part does the inverse operation of speech recognition.

First, an arbitrarily given text to be synthesized is converted to a context-dependent label sequence then an utterance HMM is constructed by concatenating the context-dependent HMMs according to the label sequence. Second, state durations of the utterance HMM are determined based on the state duration PDFs. Third, the speech parameter generation algorithm generates the spectral sequence and the excitation parameters that maximize their output probabilities. Finally, a speech waveform is synthesized directly from the generated spectral and excitation parameters using the corresponding speech synthesis filter (Mel-Log Spectrum Approximation (MLSA) filter for mel-cepstral coefficients).

At this stage study, we have used the HMM to recognize the units of the introduced text at the synthesizer input then concatenate the obtained speech units. We have used the Matlab Toolbox [5] to generate the different algorithms needed to run the HMM. The figure 2 shows a diagram that represent all steps followed on HMM-based speech synthesis system (HTS).

The preprocessor in Figure 2 consists of:

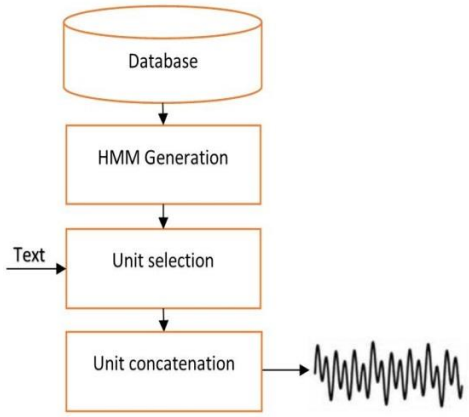

Figure 2. Proposed synthesis system

A. *Database*

The database must be large enough to contain all kinds of phonetic sequences that can appear in different linguistic contexts, which means that all speech segments are available in different prosodic forms [6]. In this study, for simplicity, we have recorded 202 sentences that contains all possible sound in the Arabic language. These recorded files made by an Arabic native speaker using cardioids microphone with a high quality flat frequency response. The speech files were sampled at 16 kHz and 16 bits; however, the database formatting was done by developing a procedure under Matlab, as follows:

- Each sentence is done as: 1.wav, 2.wav,.....,202.wav.
- For each sentence the phonetic transcription is made manually and a ”seg.text” file is generated as (1.seg, 2.seg,...202.seg.).

Each ”seg.text” file consist of the phonemes for the corresponding sentence. Its samples numbers, which characterize each phoneme of the sentence. Thus, it is possible to locate the beginning and end of each phoneme using Matlab as shown in the example below:

[15981]	[16717]	w
[16717]	[17005]	a
[17005]	[17496]	d
[17496]	[17827]	j
[17827]	[18102]	h
[18102]	[18456]	a

The impossibility of using the International Phonetic Alphabet (API) symbols under MATLAB environment, prompted us to use different types of characters (digital or other). The Table I presents the correspondence between different phonemes and their notation in the system.

TABLE I. CORRESPONDENCE BETWEEN SOME PHONEMES AND THEIR CHOSEN NOTATION

Phoneme (API)	ɸ	a	ă	ʃ	ɛ	œ
used charcters	h	a	@	c	3	q

- All possible units for similar phonemes are stored in the same folder, each folder take the name of the corresponding phoneme.
- For each folder, the unites are concatenated to obtain a single '.wav' for a phoneme.
- These different concatenated files are used to generate the HMM models.
- The database consists of the different phonemes files.

B. *HMMs Generation*

The generation of an HMM for a phoneme consists of going through the steps illustrated in Figure 3:

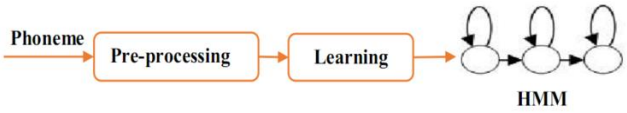

Figure 3. The steps of generating an HMM model

- Phoneme: The phoneme to be considered to generate the HMM corresponds to the concatenated one as explained during the database constitution.
- Preprocessing: The block performing this task makes the acoustic vectors as output of each input signal. These acoustic vectors will later be used as observations in the hidden Markov model. In our work, we used the ΔMFCC parameters; we have taken in account only the first 12 coefficients then adding the first derivatives ΔMFCC and the second derivatives ΔΔMFCC.
- Learning: This phase aims to generate the different HMM models adopted for each concatenated phoneme in the corpus. Indeed, to describe a phoneme by an HMM [7], it is necessary to define the topology of this HMM which represents as possible as the acoustic realizations of a phoneme. In our case, we have used 3-states HMMs. We have modeled the output in each state of the HMM by a Multi Gaussian or GMM distribution where "observations" are drawn from a Gaussian probability distribution. Instead of an observation matrix, the observations of each state are described by the mean value and the variance of a Gaussian density. GMM Learning amounts to estimate the HMM model parameters that give the best possible distribution of the acoustic vectors in each state. As illustrated in Figure 4, the overall organization chart of the learning phase.

The final stage of the learning phase is the extraction of the HMM model parameters. Each model is represented by the following variables:

- The initial probability matrix.
- The transition matrix A_revalued.
- The matrix of the mean μ_revalued

The estimated variance matrix ∑_revalued.

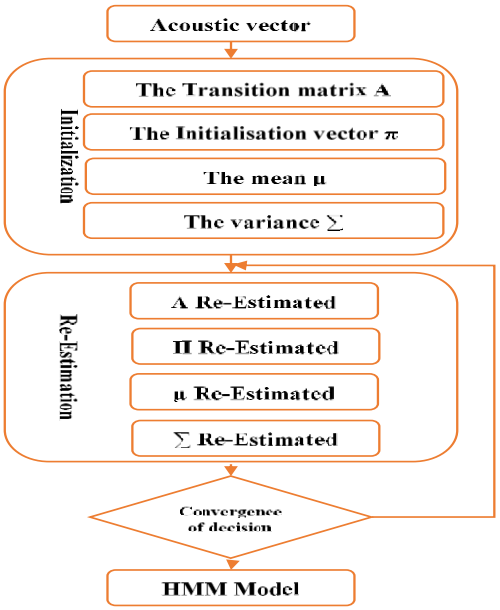

Figure 4. Flowchart of the HMM models production

C. *Unit Selection*

Once the HMM models have been prepared, the next step is to choose the best phonemes in the database that will build the output sentence, for that we apply the maximum likelihood algorithm [7], [8]. This will allow us to select the most resembling phoneme for the HMM model. For this, we compare the HMM model of each phoneme using all existing audio files, then we take the most resembles audio to the model.

1) The maximum likelihood algorithm: The likelihood calculation of the observations sequence X in relation to an HMM (λ) consists to evaluate the probability P(X/λ) [8]. The observations sequence X probability knowing the model λ is obtained by the sum of all possible states sequences probabilities:

$$P(X/\lambda) = \sum_S P\left(\frac{S}{\lambda}\right) P\left(\frac{X}{S}, \lambda\right) \quad (1)$$

In our application, we have used the Forward algorithm:

$$\alpha_t = P(x_1, \ldots, x_t, s_t = i/\lambda) \quad (2)$$

$$\alpha_1 = P(x_1, s_1 = i/\lambda) \quad (3)$$

The Forward algorithm has programmed under the Matlab environment, to calculate the likelihood of a given sequence:

for $i = 1:N$
  $\alpha_1(i) = \Pi_i b_i(x_1)$
end
for $t = 1:T-1$
  for $j = 1:N$
    $\alpha_t(j) = [\sum_{i=1}^{N} \alpha_{t-1}(i)\alpha_{ij}]b_i(x_t)$
  end
end

$$P(X/\lambda) = \sum_{i=1}^{N} \alpha_T(i)$$

To make a decision regarding the most similar phoneme to the HMM model, we have calculated the maximum of the likelihood [9].

$$Decision = \max(P(X/\lambda)) \quad (4)$$

Finally, we select for each phoneme a single audio file to synthesize the input sentence.

D. Concatenation

In order to synthesize a speech signal from an input text using the method described, we choose a sentence that we want to listen. This sentence will be written phonetically based on the symbols shown in table I. The followed steps of the HTS system can be summarized as:

- The user enters a sentence;
- For each phoneme of this sentence, the synthesizer goes through the steps of the unit selection, detailed previously;
- Then concatenates the obtained audio files.
- The user will be able to listen to the sentence he/she has entered at the synthesizer input.

III. Results and Discussions

The first step of the method consisted in the realization of the phonemes database from the initial corpus. The Table II represents some phonemes existing in the database and their occurrence numbers.

TABLE II. Phonemes with Their Number of Occurrences

Phonemes	@	a	b	c	d	f	h	i	k	j
Number of occurrences	10	39	22	20	15	10	15	29	22	10
Phonemes	l	m	n	w	T	r	s	t	X	y
Number of occurrences	35	10	20	14	4	12	26	18	8	14

As an example, we have considered the different steps in the generation of HMM for the phoneme /f/

1) Preprocessing results: We consider for the preprocessing the concatenated phoneme file /f/. We calculate its acoustic vector which will correspond to the observation vector for the /f/ HMM.

$$observation = \begin{bmatrix} -7.3272 \\ -0.3011 \\ -0.7261 \\ 0.4224 \\ -0.7020 \\ -0.3997 \\ -0.5919 \\ -0.3032 \\ -0.1425 \\ -0.3673 \\ -0.1511 \\ -0.1342 \end{bmatrix}$$

2) Learning results: A self.mat file was created automaticly which contain all HMM mode parameters. Each HMM model will then be stored as an object with the fields self.mu, self.Sigma, self.transmat and self.prior.

- The means fields (μ) contain a mean matrix vectors, where each column of the matrix corresponds to a state of the HMM;
- The Sigma fields contain three-dimensional matrix of matrices of variance, where the third dimension corresponds to the state;
- The transmittable fields contain the transition matrix;
- The prior field contains the initial matrix probabilities for each state;

As shown below:

self =

nState:	3
nMixt:	2
prior:	[3x1 double]
transmat:	[3x3 double]
mixmat:	[3x2 double]
mu:	[12x3x2 double]
Sigma:	[4-D double]

The initial probabilities matrix (π), transition (A), and averages (μ) for the model HMM of a phoneme /f/ (HMMf) are given as follows:

- Probabilities matrix (π):

$$\pi = [1 \quad 0 \quad 0]$$

- Transition matrix (A):

$$A = \begin{bmatrix} 0.7424 & 0.1316 & 0.1260 \\ 0.0721 & 0.8589 & 0.0690 \\ 0.1678 & 0.1343 & 0.6979 \end{bmatrix}$$

- Average matrix of the first Gaussian:

$$\mu = \begin{bmatrix} -7.5297 & -9.6022 & -1.4663 \\ -0.6931 & -1.1182 & 0.1394 \\ 0.3605 & -0.7646 & -0.6160 \\ 0.9962 & 0.0443 & 0.0108 \\ -0.8747 & -0.7937 & -0.6840 \\ -0.4428 & -0.7230 & -0.4810 \\ -0.2720 & -0.7051 & -0.6317 \\ 0.1611 & -0.0717 & -0.6914 \\ -0.1275 & 0.1409 & -0.3353 \\ -0.7569 & -0.1934 & -0.1169 \\ -0.5303 & 0.0163 & 0.2124 \\ -0.1798 & 0.1206 & -0.2234 \end{bmatrix}$$

- Average matrix of the second Gaussian:

$$\mu = \begin{bmatrix} -8.9434 & -6.3812 & -6.0680 \\ -0.0053 & -0.2922 & 1.0378 \\ -1.0274 & -1.4150 & -0.0930 \\ 1.2243 & -0.2711 & 1.0597 \\ -0.3900 & -1.0913 & -0.1624 \\ 0.1062 & -0.6294 & 0.0567 \\ -0.3491 & -0.9387 & -0.3689 \\ -0.3030 & -0.6978 & -0.2487 \\ -0.0853 & -0.3921 & -0.2857 \\ -0.3962 & -0.3453 & -0.5953 \\ -0.2324 & -0.1239 & -0.3960 \\ -0.3594 & -0.0180 & -0.4456 \end{bmatrix}$$

3) Unit selection results: The likelihood calculation, gives as a result a vector which contains the different values of thelikelihood with respect to each phoneme, then we take the maximum of these values. The table III gives the results of the log likelihood calculation (ll) for the phoneme /f/.

TABLE III. THE LIKELIHOOD LOGARITHM FOR THE PHONEME /F/

Phonemes	f1010	f21	f52	f53	f64
ll	247.9	197.1	148.6	200.8	129.5
Phonemes	f65	f66	f77	f78	f99
ll	-274.1	-180.1	-234.6	-163.9	-239

From this table, we see that ll_{max}=129.5 so the synthesizer will choose the file "f64.wav" as representative of the phoneme /f/ at the output among the 10 files that are in the database.

This operation is applied for all phonemes in the database. At the output of the synthesizer, we have 33 audio files that will build a sentence of our choice.

4)Concatenation Results: As an example, we have chosen the color red (Ahmar). Its transcription was done by the system based on the table I is: "AHmar". After passing through the step of unit selection, and concatenate the selected phonetics elements, we could listen to the word we have witten. To assess the intelligibility and natural aspects of the obtained synthesis voice, we applied two types of tests. In these tests, we randomly selected a group of people to evaluate the speech of the developed system. The participant are 15, with different professions and knowledge of the Arabic language in order to get a good assessment purposes. The first test which measures the intelligibility is divide in two sub-test. In the first part, each participant listen to a sentence. Then marks on an answer sheet (four choices) which sentence is listen for (Test 1A). In the second part, ten sets of sentences are chosen, each one has four sentences. The sentences differ only in a single consonant on its words for the same set. The listeners are asked to mark on the answer sheet which number correspond to the written sentence (Test 2A). In the second test (quality), we evaluated the system with MOS and PESQ tests.

a) Mean Opinion Score (MOS): The Mean Opinion Score (MOS) provides a numerical indication of the perceived quality of received media after compression and/or transmission. The MOS is expressed as a single number in the range 1 to 5, where 1 is lowest perceived audio quality, and 5 is the highest perceived. The listeners were asked a few questions about several attributes such as the speed, the pronunciation, the stress of the speech and they were asked to rank the voice quality using afive level scale (5 - Excellent 4 - Very good, 3- good, 2 – Average, 1 - bad).

b) Perceptual Evaluation of Speech Quality (PESQ): The PESQ is an objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs. It compares the original signal with the corresponding (degraded) synthesis signal.

The final averages of the result's tests for the MOS and PESQ are 3.2, 2.5 respectively.

IV. CONCLUSION

This study focused on the contribution to the development of the HMM-based speech synthesis system (HTS) for the Arabic language. We started with the implementation of the database, which requires special care in relation to its richness in order to constitute the linguistic units necessary for a good quality of the synthetic signal. The second important module in a HTS system is the language analysis and processing module, which is related to the language used and whose role is crucial in order to remove many ambiguities that may compromise the meaning of the synthesized sentence.

The speech units modeling by HMM also plays a very important role in selecting the best speech unit. Indeed, the choice of the Markov model parameters is a paramount importance in the result obtained as for example the number of states chosen, the constituted observation vectors or the transition matrices of the model. The last step is the concatenating the different speech units resulting from the HMM treatments. At this stage, the problems encountered are linked to many points: the discontinuity at the ends of concatenated units which require special treatment to make the synthesized signal more fluid. The choice of the synthesis model (here Straight [10], [11], [12]) to join the prosody of each obtained synthetic sentence in order to give the necessary expressiveness and naturalness. The obtained results are satisfied and the overall intelligibility of the system was considered acceptable by the auditors. The contribution has enabled us to show that all these points raised, which compromise the obtained speech synthesis quality may deserve to be dwelt on. In order to find rules that will bring a plus to the system in terms of speech synthesis quality, and to apply the method for an expressive Arabic Text-to-speech.

REFERENCES

[1] S. Arik, G. Diamos, A. Gibiansky, J.Miller, K. Peng, W. Ping, J. Raiman,Y. Zhou,'Deep Voice 2: Multi-Speaker Neural Text-to-Speech',may,2017.

[2] A. Amrouche, L. Falek, H. Teffahi, 'Design and Implementation of a Diacritic Arabic Text-To-Speech System', International Arab Journal of Information Technology (IAJIT). Volume 14, No. 4, July 2017.

[3] H. Zen, T. Nose, J. Yamagishi, S. Sako, T. Masuko, A. W. Black, and al., 'The HMM- based speech synthesis system (HTS) version 2.0,' in SSW, pp. 294-299, 2007.

[4] J .Yamagishi, 'An introduction to hmm-based speech synthesis,' Technical report, Tokyo Institute of Technology 2006.

[5] K. Murphy, 'Hidden Markov Model (HMM) Toolbox', 1998. Http://www.ai.mit.edu/ murphyk/Software/hmm.html

[6] M. Boudraa, B.Boudraa, B. Guerin, 'Elaboration d'une base de donnes arabe phontiquement quilibre', Actes du colloque langue Arabe et Technologies Informatique Avences, pp 171-187, Casablanca, December 1993.

[7] F. Takahashi, T. Masuko, K. Tokuda, and T. Kobayashi, 'A study on performence of a very low bit rate speaker independent HMM vocoder' ASJ Spring meeting, 2-P-23, pp.313-314, Mars. 1999.

[8] A. Abed, Système d'Aide Orthophonique à la Substitution Phonémique Infantile Basé sur les HMM/GMM, Ecole Nationale Polytechnique. Thèse de Doctorat, April 2017.

[9] Z. Heiga and al. 'The HMM-based Speech Synthesis System (HTS)' Version 2.0.http://www.sp.nitech.ac.jp.

[10] H. Kawahara, 'Speech representation and transformation using adaptive interpolation of weighted spectrum: vocoder revisited,' in Acoustics, Speech, and Signal Processing, 1997. ICASSP-97., 1997 IEEE International Conference on, 1997, pp. 1303-1306.

[11] H. Kawahara, I. Masuda-Katsuse, and A. de Cheveign, 'Restructuring speech representations using a pitch-adaptive timefrequency smoothing and an instantaneous-frequency-based F0 extraction: Possible role of a repetitive structure in sounds,' Speech communication, vol. 27, pp. 187-207, 1999.

[12] H. Kawahara, 'STRAIGHT, exploitation of the other aspect of VOCODER: Perceptually isomorphic decomposition of speech sounds,'Acoustical science and technology, vol. 27, pp. 349-353,2006.