Le traitement automatique de l'arabe, JEP–TALN 2004, Fès, 19–21 avril 2004

An Arabic TTS System Based on the IBM Trainable Speech Synthesizer

Amr Youssef (1,2), Ossama Emam (2)
(1) Department of Electronics and Communications Engineering
Cairo University, Giza, Egypt
ayoussef@eg.ibm.com
(2) Human Language Technologies Group, IBM, Egypt Branch
P.O. Box 166 El-Ahram, Giza, Egypt
emam@eg.ibm.com

Mots-clefs – Keywords

Système de passage du Texte-à-Parole Arabe
Arabic Text-to-Speech, IBM Concatenative Speech Synthesizer

Résumé - Abstract

En cet article, nous donnons une vue d'ensemble du système du Texte-à-Parole Arabe que nous avons développé au laboratoire de technologies de langue humaine d'IBM Egypte. Ce système est basé sur le dernier cri d'IBM synthétiseur de parole trainable. Un bref revue des composants principaux du système est présenté avec une certaine emphase sur les dispositifs qui sont appropriés à la langue arabe. En conclusion, des points moyens d'opinion pour le discours synthétisé sont présentés.

In this paper, we give an overview of the Arabic Text-to-Speech (TTS) system that we developed at the Human Language Technologies laboratory of IBM Egypt. This system is based on the state of the art IBM trainable concatenative speech synthesizer. A brief review of the major system components is presented with some emphasis on the features that are relevant to the Arabic language. Finally, a mean opinion score for the synthesized speech is presented.

1 Introduction

Most of the existing commercial speech synthesis systems can be classified as either formant synthesizers (Klatt, 1980), (Styger et. al, 1994) or concatenation synthesizers (Donovan, 1996), (Hamza, 2000). Formant synthesizers, which are usually controlled by rules, have the advantage of having small footprints but the synthesized speech usually doesn't sound so natural. On the other hand, trainable concatenative speech synthesis, using large speech databases, has become popular due to its ability to produce high quality natural speech output. The large footprints of these systems do not present a practical problem for applications where the synthesis engine runs on a server with enough computational power and sufficient storage.

Although the area of Arabic Text-to-Speech (TTS) is still in its infancy, compared to other languages such as English, there are currently several commercially available Arabic TTS systems such as ARABTALK, BrightSpeech, ElanSpeech and Sakhr TTS available from Research and Development International (RDI), Babel Technologies and Babel-Infovox, ElanSpeech, and Sakhr software respectively.

In this paper, we describe the current status of the Arabic TTS system that we developed at the Human Language Technologies laboratory of IBM Egypt. This system is constructed using the state of the art IBM trainable unit-selection based concatenative speech synthesizer described in (Eide et. al, 2003), (Donovan et. al, 2001), (Donovan et. al, 1998).

The paper is organized as follows. In the next section, we describe the phonetic set, word syllabification and syllabitic stress rules for the Arabic language. In section 3, we give an overall description of the system. In section 4, we describe the system construction procedure. In section 5, the run time synthesis process is briefly reviewed. The result of our subjective testing is given in section 6. Finally, a brief discussion and future work are given in section 7.

2 Arabic Phonetic Set and Word syllabification Rules

With very few exceptions, letter to sound conversion for Arabic usually has simple one to one mapping between orthography and phonetic transcription for given correct diacritics.

Figure 1 shows the places and manner of articulation for the list of Arabic consonants used by our system [1]. The current system uses 14 vowels to accommodate for short and long vowels as well as for the emphatic vowels. Although a shorter list of vowels yields a better performance in our current Arabic automatic speech recognition engine, our experiments showed that the acoustic trees were not always able to distinguish between some normal and emphatic vowels based on the phonetic context (An example for this case is the word "Baba" (Arabic word for father) and the word "Bab" (Arabic word for door)). For this reason, we preferred to explicitly represent these vowels.

The syllabic structures in Arabic are limited in number and easily detectable. Every syllable begins with a consonant followed by a vowel which is called the nucleus of the syllable. Short vowels are denoted by "V" and long vowels are denoted by "$V:$". It is obvious that the vowel exists in the second place of the syllable. These features facilitate the process of syllabification. We can classify the syllables in Arabic either according to the length of the syllable or according to the end of the syllable as follows:

CV	$short$;	$open$.
CV:	$long$;	$open$.
CVC	$long$;	$closed$.
CV:C	$long$;	$closed$.
CVCC	$long$;	$closed$.
CV:CC	$long$;	$closed$.

[1] In here, we are using the Speech Assessment Methods Phonetic Alphabet (SAMPA) notation

PLACE OF ARTICULATION

MANNER OF ARTICULATION			Bilibial	Libio-Dental	Inter-Dental Alveo-Dental Alveolar	Palatal	Velar	Uvelar	Pharangeal	Glottal
Stop (Plosive)	Voiced	Pharyngealized			ض d`					
			ب b		د d	ج g				
	Unvoiced	Pharyngealized			ط t`			ق q		
					ت t		ك k			ء ?
Fricative	Voiced	Pharyngealized			ظ D`					
					ذ D	ز z			ع ?	
	Unvoiced	Pharyngealized			ص s`					
				ف f	ث T	س s	ش S	خ* x	ح X\	ه h
Affricate	Voiced					ج Z				
Nasal	Voiced		م m		ن n					
Trill	Voiced				ر* r					
Lateral	Voiced				ل* l					
Semivowel	Voiced		و w			ى j				

*** In some rare cases may act as pharyngealized**

Figure 1: List of Arabic Consonants

The first four types may occur in any position of the word. The last two types occur at the end of the word or alone as a single word.

The syllables in a single word are not pronounced in the same level of loudness. It is possible to find three different levels of loudness in the same word. The only cause of these differences is what is called the stress. Stress is the intensity (energy) of the time domain signal of the syllable. The degree of the stress are three: primary (main) stress, secondary stress and tertiary (weak) stress. The location of the different types of the stress in the word depends on the types of the syllables, their distribution and their numbers.

- When the word consists of sequence of short CV syllables, the first syllable will get the main stress and the rest of the syllables will gain the weak stress.

- When the word contains one long syllable, this syllable will take the main stress, and the remaining will receive the weak stress.

- When the word contains two or more long syllables, the long one which is nearest to the end of the word (not the last one) will receive the primary (main) stress, and the long syllable which is near to the beginning of the word (not the first one) will take the secondary stress.

In our system, as will be explained in the next section, the syllable stress is included in the set of features that are used to predict the target pitch by the prosody module.

3 Overall System Description

Figure 2 shows the architecture of the current system. It is composed of three major components: a text module, a prosody module, and a back-end module. The text module includes a text normalizer, phonological analyzer, and a prosodic planner. The text analyzer does standard text normalization tasks such as converting digits into their words equivalent, and spelling out some known abbreviations. The Arabic phonological analyzer is responsible for grapheme to phoneme transformation (rule based with the option of activating an exception dictionary), syllabification, and syllable stress assignment. The prosodic planner does some abstract prosodic planning at the text level. The decision to rely on manual diacritization (and not to consider the diacritization module as a part of the IBM Arabic TTS system) is based on the fact that the current state of the art automatic Arabic diacritization techniques are not mature enough. In fact, none of the tested commercial automatic diacritization tools provided the quality required by our current TTS client. The prosody module generates pitch, duration and energy targets and the back-end searches a large speech database to select segments that minimize a cost function, concatenate them and performs signal processing on the resulting synthesized speech.

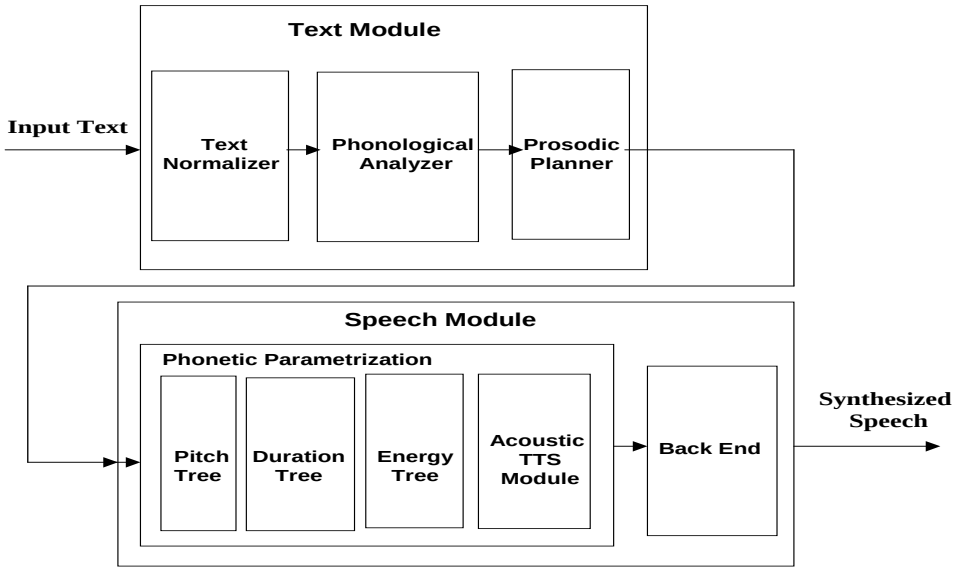

Figure 2: Overall System Architecture

4 System Construction

The system construction procedure can be divided into two main tasks. The first one is the raw speech database preparation and the second is the synthesis database build. The raw speech database preparation includes:

- The Arabic text script preparation.
- The speech database recording.

The synthesis database build procedure includes:

- Signal processing.

- Hidden Markov Models (HMMs) alignment.
- The (acoustic, energy, pitch and duration) trees build.
- Optional acoustic database preselection process.

Figure 3 shows an overview of the system construction.

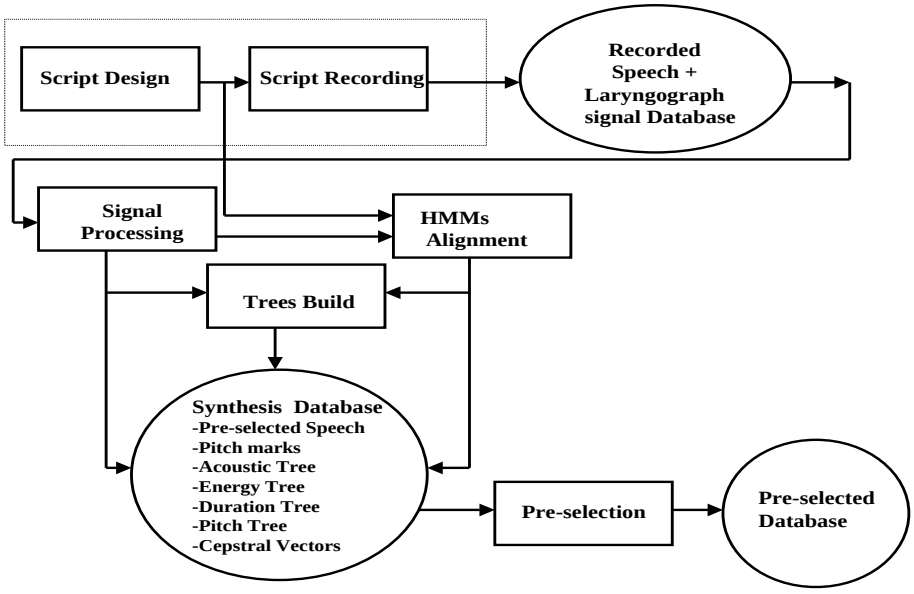

Figure 3: System Construction Procedure

4.1 Speech Database Preparation

The first step in the dataset preparation is to prepare the script to be read during recording. The first 1000 sentences of the script (which compromise about 40% of the overall recorded speech database) were obtained using a greedy optimization algorithm to optimize the phonetic balance of the recorded script. This algorithm computes a score for every sentence not yet selected based on the diphones within it and the wider contexts in which they occur. The wider context comprises the diphone, the preceding phone and any word boundaries between the three phones. The score is heavily weighted in favor of diphones, in previously unseen wider contexts, which are under-represented in the script selected to date. The best scoring sentence is added to the script and the process repeated. The result is a training script in which the most diverse sentences are at the top, and in which every additional sentence brings as much new variety in phonetic context as possible.

Due to the lack of a large well diacritized Arabic corpora, a decision was made to use undiacritized text in this process. A special phonetic set for the undiacritized text was designed and the 1000 sentences were obtained by running the greedy algorithm over about $10,000$ Modern Standard Arabic (MSA) [2] sentences obtained from different Arabic sites on the web.

[2] Arabic is usually classified as classical Arabic, Modern standard Arabic and colloquial Arabic. Classical Arabic is the language of the Quran, the Islamic tradition and the great writers and poets. The modern classical Arabic is the form of the Arabic that is taught in schools and is used in the medias, press, formal talks and most broadcast stations. Colloquial Arabic (dialects) in different Arabic speaking countries differ from each other, even within the same country. Modern Standard Arabic is generally adopted as the common medium of communication through the Arab world at present time. Therefore we choose the modern standard Arabic literary Arabic spoken in Egypt as our model of pronunciation.

In addition to the 1000 phonetically-balanced sentences, an additional 2000 sentences were taken from domains of interest to our current clients.

The recordings were done in a professional recording studio using a Neuman U87A microphone in a sound-proof room. Although the system build process requires speech sampled at 22KHz, the speech data base was collected at a 48kHz for future use. The speech recording is made in stereo with a laryngograph signal recording.

During the recording process, the professional speaker was given an undiacritized text script. After the recording, the script was manually diacritized in accordance to the speaker readings. The diacritized script was manually reviewed twice in order to make sure that it matches the recorded speech. Throughout this process, the phrase marks (such as question marks, periods, and commas) were also added.

4.2 The Synthesis Database Build

The overall 3000 sentences compromise about 11 hours of the professional speaker speech (including silence). The synthesis database build procedure converts this recorded speech into a synthesis database that can be used by the synthesizer backend during runtime. In what follows, we describe the four basic steps involved in this procedure.

4.2.1 Signal Processing

The recorded speech data is initially coded into 12 dimensional mel frequency cepstral coefficients (MFCCs) plus log energy and the first and second time differentials of these parameters using 25ms frames at a uniform 10ms frame rate.

Pitch marks were produced using the Wavelet transform approach (Sakamoto et. al, 2000). At first, we produced the pitch marks using the glottal closure signal obtained from the professional speaker during the recording. However, the final system was build utilizing the marks obtained using the Wavelet approach since it yields better synthesized speech.

The data is recoded using 25ms Pitch Synchronous (PS) frames through regions of voiced speech, and 6ms frames at a 3ms frame rate through unvoiced speech, using the scaling method described in (Donovan, 1996) to ensure that cepstra from the different sized frames are comparable. The recoding gives better resolution through regions of unvoiced speech which is especially helpful in segmenting plosives due to the short timescales and rapid transitions involved.

4.2.2 HMM Alignment

The TTS system text module is used to prepare a pronunciation dictionary for the words in the script. A set of speaker independent HMMs (Rabiner, 1989) is used to obtain an initial phonetic alignment of the speech.

This initial alignment is used to train a set of speaker-dependent decision-tree (Cherkassky et al, 1988) state-clustered HMMs (Bahl et al, 1993). These HMMs are mostly 3 state left-to-right models, except for the voiced plosive models which align better with 2 states.

After further training the HMMs are used to provide another alignment, and the model building process repeated using the PS recoded data. After more training the final models are used to provide an HMM

feneme-level [3] alignment of the training data.

Perceptually modified cepstral vectors and spectral continuity cost decision trees are also computed from the data and the final HMM alignment. These components are used to determine spectral continuity costs between segment endpoints during the runtime search.

4.2.3 Acoustic, Energy, Pitch and Duration Trees Build

The acoustic decision trees used in synthesis are built from the final HMM alignment. A separate decision tree is built for each unclustered HMM state using the standard maximum likelihood tree growing procedure used in most modern speech recognition systems (Bahl et al, 1993). The splits are made using broad class context questions applied to the immediate phonetic and word boundary context only, with the likelihood computations based on the MFCC, energy and time differential parameters described above. The criteria used to stop tree growth are to insist on a minimum number of speech segments per leaf and a minimum increase in log-likelihood per split.

The current settings resulted in an acoustic tree with about 40K leaves corresponding to about 1 Mega non-silence speech segments.

Energy prediction decision trees are also built from the final HMM alignment. Again, a separate tree is built for each unclustered HMM state using the same tree growing procedure.

Pitch and pitch deltas prediction decision trees are built using the approach described in (Eide et. al, 2003). End pitch and delta pitch for each syllable are predicted from a set of features gathered from the text. These features include the phrase type, lexical stress of the current syllable, and distance of the current word from the beginning of the current phrase and from the end of the phrase. Details of the pitch target estimation is given in (Eide et. al, 2003). The only difference for the Arabic system is that the Part of Speech (POS) and the phrase level stress of the current word are not yet included in the feature vectors.

The latest IBM TTS system (Eide et. al, 2003) constructs the decision tree to estimate the target duration from a set of features similar to that used to estimate the target pitch. Again, the only difference for the Arabic system is that the Part of Speech (POS) and the phrase level stress of the current word are not yet included in the feature vectors.

4.2.4 Acoustic Database Preselection Process

In order to reduce the runtime system size and improve runtime speed some of the training data is discarded from the runtime dataset. This preselection is performed by keeping only the most commonly used segments using the data driven technique described in (Hamza et. al, 20002).

5 Run Time Synthesis

During the synthesis process, the words to be synthesized are normalized before they are passed to the automatic phonetic transcription module. The decision trees are used to convert the phone sequence into an acoustic, pitch, duration, and energy leaf for each feneme in the sequence.

[3] A feneme is a term used to describe individual HMM model positions, e.g., the model for the phone $/L/$ compromises three fenemes L_1, L_2 and L_3.

The next stage of the synthesis is to perform dynamic programming search over all waveform segments aligned to each acoustic leaf to determine the segment sequence which minimize the cost function. A Viterbi beam search, in which all the candidates in the target leaf as defined by the decision tree, are considered. For speed, the backing off procedure described in (Donovan et. al, 1998) is no longer used, i.e., segments from other contexts are no longer considered. All the segments in the target leaf are scored using the best cost among the cost computed from all possible predecessors' costs to date and the pitch transition and spectral transition cost to them. All segments within twice the beam width of the best segments are then scored against the target prosody. Segments within the beam width of the best segments are then retained for use in the next step in the dynamic programming forward pass. After this stage, an energy discontinuity smoothing (Donovan et. al, 1998) and pitch discontinuity smoothing (Eide et. al, 2003) are performed. The signal processing routines concatenate the selected segments and modify them to have the target prosody, or the capped prosodic values (Donovan et. al, 1998) if applicable. The modification algorithm is similar in concept to the frequency domain PSOLA algorithm described in (Moulines et. al, 1990).

6 Results of Subjective Testing

In order to assess the quality of our current system, a subjective test was performed. A set of five news sentences was used as the test material. These sentences were not in the training script of our system. The test sets were played to fifteen volunteer listeners (the 1^{st} five are females and the rest are males) who were asked to rate the system intelligibility, naturalness and overall voice quality on a scale of 1 (bad) to 5 (good). The volunteer listeners were asked to give the rating based on how good they thought the sentences were without any further definition of the word "good". The listeners were encouraged to use the full five point scale. The obtained test result is shown in Figure 4. The average scores obtained are $4.0, 3.4$, and 3.65 for intelligibility, naturalness and overall voice quality respectively. Under the same conditions, when the same test was performed on the synthesized speech produced by the other Arabic TTS demo systems available on the web, the best system (name is omitted on purpose) obtained $3.35, 2.6$, and 2.7 for intelligibility, naturalness and overall voice quality respectively. Using the paired two sample t-Test for means analysis, our system overall voice quality was better than the best tested system at 0.008 level.

The synthesized speech samples produced by our system for this experiment will be played out during the conference presentation.

7 Discussion and Future Work

The lack of large well-diacritized and POS tagged Arabic corpora is still one of the obstacles in the Arabic TTS building process. Such corpora would have been used both at the initial recording script preparation and at the preselection process (Hamza et. al, 20002). It would have also been used to train a statistical POS tagger that may lead to a better target pitch and target duration prediction. The availability of such a database will certainly help in the development process of Arabic TTS systems both at the industrial and at the research institutes. Improving the HMM models used in the automatic alignment process will contribute to the improvement of the Arabic TTS system. Manual alignment (and alignment checking) is not an easy task since the IBM system is based on fenemes. Improving the pitch marks prediction technique will certainly help in improving the quality of the synthesized speech.

We are currently working on the development of a male version of the system with a larger speech database (about 15 hours). This version will include a larger list of consonants (e.g., G as in "technolo**G**y" and v as in "**v**ienna").

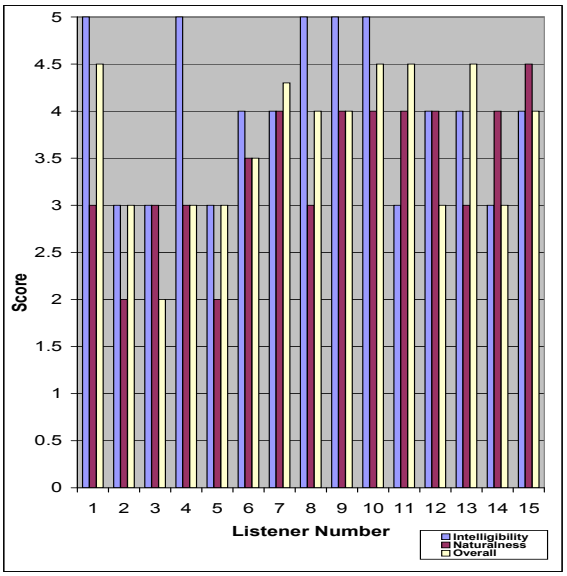

Figure 4: Mean Opinion Score Test Results

References

Bahl L.R, deSouza P.V, Gopalakrishnan P.S, Picheny M.A (1994), Context Dependent Vector Quantization for Continuous Speech Recognition, *Proc. of ICASSP' 93*.

Cherkassky V., Mulier F. (1988), *Learning from Data: Concepts, Theory and Methods*, John Wiley & sons, inc.

Eide E., Aaron A., Bakis R., Cohen P., Donovan R., Hamza W., Mathes T., Picheny M. , Polkosky M., Smith M., Viswanathan M. (2003), Recent Improvements to the IBM Trainable Speech Synthesis System, *Proc ICASSP'03*, Hong Kong, China.

Donovan R.E. , Ittycheriah A., Franz M., Ramabhadran B., Eide E., Viswanathan M., Bakis R., Hamza W., Picheny M., Gleason P., Rutherfoord T., Cox P., Green D. , Janke E., Revelin S., Waast C., Zeller B., Guenther C., Kunzmann J. (2001), Current Status of the IBM Trainable Speech Synthesis System, *Proc. 4th ESCA Tutorial and Research Workshop on Speech Synthesis*, Scotland, UK.

Donovan R.E. (1996), Trainable Speech Synthesis, *PhD. Thesis*, Cambridge University, Engineering Department.

Donovan R.E., Eide E.M (1998), The IBM Trainable Speech Synthesis System, *Proc. ICSLP'98, Sydney, Australia.*

Hamza W. (2000), Arabic Speech Synthesis Using Large Speech Database, *PhD. Thesis*, Cairo University, Electronics and Communications Engineering Department.

Hamza W., Donovan R.E. (2002), Data-Driven Segment Preselection in the IBM Trainable Speech Synthesis System, *Proc ICSLP'02*, Denver, CO, USA.

Klatt D.H. (1980), Software for a cascade/parallel formant synthesizer, *Journal of the Acoustical Society of America*, VOL 67, 971-995.

Moulines E., Charpentier F. (1990), Pitch-Synchronous Waveform Processing Techniques for Text-to-Speech Synthesis Using Diphones, *Speech Communication*, VOL 9, 453-467.

Rabiner L. (1989), A tutorial on Hidden Markov Models and selected applications in speech recognition, *Proc. IEEE*, VOL 77(2), 257-286.

Sakamoto M., Saito T. (2000), An Automatic Pitch Marking, Method Using Wavelet Transform, *Proc. ICSLP'00*, Beijing, China.

SAMPA, *Speech Assessment Methods Phonetic Alphabet, http://www.phon.ucl.ac.uk/home/sampa/home.htm*.

Styger T., Keller E. (1994), *Fundamentals of Speech Synthesis and Speech Recognition: Basic Concepts, State of the Art, and Future Challenges Formant synthesis*, In Keller E. (ed.), 109-128, Chichester: John Wiley.