GENERATION OF SYNTHETIC SPEECH FROM TURKISH TEXT

Haşim Sak[a], Tunga Güngör[a], and Yaşar Safkan[b]

[a] Department of Computer Engineering, Boğaziçi University
Bebek, 34342, İstanbul, TURKEY
phone: +(90) 212 3597094, fax: +(90) 212 2872461, email: hasim@gvz.com.tr, gungort@boun.edu.tr
[b] Department of Computer Engineering, Yeditepe University
Kayışdağı, 81120, İstanbul, TURKEY
phone: +(90) 216 5780420, email: ysafkan@cse.yeditepe.edu.tr

ABSTRACT

In this paper, we design and develop an intelligible and natural sounding corpus-based concatenative speech synthesis system for Turkish. The implemented system contains a front-end comprised of text analysis, phonetic analysis, and optional use of transplanted prosody. The unit selection algorithm is based on commonly used Viterbi decoding algorithm. The back-end is the speech waveform generation based on the harmonic coding of speech and overlap-and-add mechanism. In this study, a Turkish phoneme set has been designed and a pronunciation lexicon for root words has been constructed. For assessing the intelligibility of the synthesized speech, a DRT word list for Turkish has been compiled. The developed system obtained 4.2 Mean Opinion Score (MOS) in the listening tests.

1. INTRODUCTION

Speech synthesis is the process of converting written text into machine-generated synthetic speech. Concatenative speech synthesis systems form utterances by concatenating pre-recorded speech units. In corpus-based systems, the acoustic units of varying sizes are selected from a large speech corpus and concatenated. The speech corpus contains more than one instance of each unit to capture prosodic and spectral variability found in natural speech; hence the signal modifications needed on the selected units are minimized if an appropriate unit is found in the unit inventory. The use of more than one instance of each unit requires a unit selection algorithm to choose the units from the inventory that match best the target specification of the input sequence of units.

ATR v-Talk speech synthesis system developed at ATR laboratories introduced the unit selection approach from a large speech database [1]. The selection of units was based on minimizing an acoustic distance measure between the selected units and the target spectrum. In CHATR speech synthesis system, prosodic features like duration and intonation have been added to the target specification to choose more appropriate units [2]. Hunt and Black have contributed to the area the idea of applying Viterbi decoding of best-path algorithm for unit selection [3]. The Next-Gen speech synthesis system developed at the AT&T laboratories is one of the commercial systems that use unit selection [4]. The front-end, i.e. the text and linguistic analysis and prosody generation is from Flextalk, the unit selection is a modified version of CHATR, and the framework for all these was borrowed from the Festival. As an improvement to the CHATR unit selection, the system uses half phones compared to phonemes as the basic speech units [5]. For the back-end, a Harmonic plus Noise Model (HNM) representation of the speech has been developed [6]. Unit selection based concatenative speech synthesis approach has also been used in the IBM Trainable Speech Synthesis System [7]. The system uses the Hidden Markov Models (HMMs) to phonetically label the recorded speech corpus and aligns HMM states to the data. The units used in the unit selection process are HMM state sized speech segments. The unit selection is a dynamic programming based search, which uses decision trees to facilitate the choice of appropriate units, with a cost function to optimize. The segments in the speech database are coded into Mel-Frequency Cepstrum Coefficients (MFCCs).

In this paper, we propose an intelligible and natural sounding corpus-based speech synthesis system for Turkish which is an agglutinative language and has a highly complex morphological structure. The research in this paper is directed towards agglutinative languages in general and Turkish in particular. In this study, we take the special characteristics of Turkish into account, propose solutions for them, and develop a speech synthesis system for the language.

2. SYSTEM ARCHITECTURE

The architecture of the system is shown in Figure 1. The components shown are common in most of the speech synthesis systems that use unit selection. The system can be mainly divided into three parts: analysis (front-end), unit selection, and generation (back-end). The analysis module is responsible for producing an internal linguistic and prosodic description of the input text. This description is fed into the unit selection module as the target specification. The unit selection module uses this specification to choose the units from the speech database such that a cost function between the specification and the chosen units is minimized. The waveforms for the selected units are then concatenated in the generation module, where the smoothing of concatenation points is also handled.

The speech corpus used for testing the algorithms developed in this research contains about 20 hours of speech recorded by a professional female speaker covering about 30000 Turkish phrases. It has been divided into two sets: training set and test set. The test set contains 1000 phrases used for the purpose of evaluating the synthesis quality. From the remaining recordings (training set), two speech unit inventories of different sizes have been constructed. One contains all the recordings in the training set (about 19 hours of speech) and the other contains 5000 phrases (about 3 hours of speech). The use of two training sets of different sizes enables us to observe the effect of the corpus size on the output quality and on the performance of the algorithms.

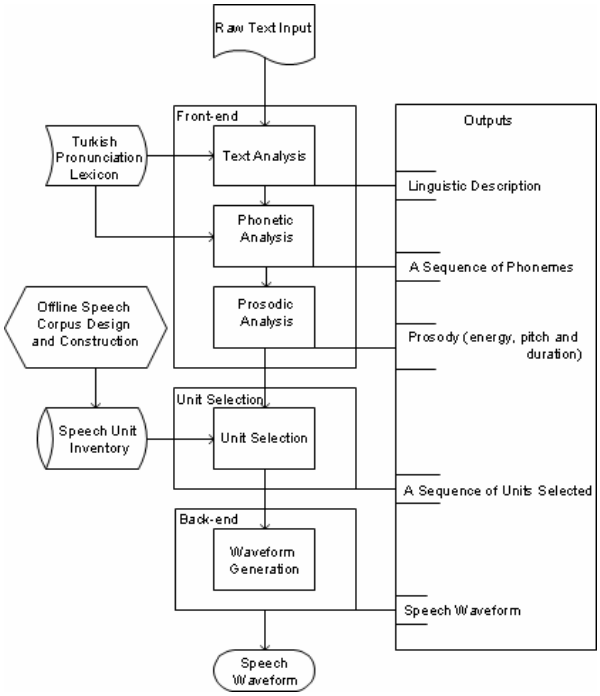

Figure 1. Corpus-based concatenative Turkish speech synthesis system architecture

3. FORMING LINGUISTIC AND PROSODIC DESCRIPTION

In a language, phonemes are the smallest units of sound that distinguish one word from another [8]. Turkish alphabet contains 29 letters classified as 8 vowels (a, e, ı, i, o, ö, u, ü) and 21 consonants (b, c, ç, d, f, g, ğ, h, j, k, l, m, n, p, r, s, ş, t, v, y, z). However, Turkish orthography cannot represent all the sounds in Turkish. In this study, for phonetic transcriptions we used a phoneme set based on the SAMPA phonetic alphabet [9]. The SAMPA identifies 8 vowels and 24 consonants (excluding two consonantal allophones /w/ of /v/ and /N/ of /n/) for representing Turkish sounds and designates a length mark /:/ to represent the lengthening of some vowels in loanwords in Turkish.

A Turkish lexicon has been built containing about 3500 root words and their pronunciations. The lexicon is used to determine the pronunciations of the words and to expand the abbreviations and acronyms. The small size of the lexicon is because of the relatively simple pronunciation schema of Turkish compared to English. Turkish is a phonetic language in the sense that a simple grapheme-to-phoneme conversion (i.e. one-to-one mapping of letters to phonemes) is possible for most of the words due to the close relationship between orthography and phonology. Most of the words in the lexicon are those for which such a direct mapping cannot yield the correct pronunciation due to vowel lengthening, palatalization, etc., and most of them are loanwords originated from languages like Arabic and Persian [10].

Turkish is an agglutinative language – given a word in its root form, we can derive a new word by adding an affix (usually a suffix) to this root form, then derive another word by adding another affix to this new word, and so on. This iteration process may continue several levels. A single word in an agglutinative language may correspond to a phrase made up of several words in a non-agglutinative language. Thus, the text should be morphologically analyzed in order to determine the root forms and the suffixes of the words before further analysis [11, 12]. We used a morphological analyzer based on Augmented Transition Network (ATN) formalism [12]. The root word pronunciations are then looked up in the lexicon. If a root word cannot be found in the lexicon, the pronunciation is formed by a direct mapping of letters to phonemes in the phoneme set. This is also the case for suffixes: the pronunciations of all suffixes are formed in a direct manner. In this study, no linguistic analysis on syntax and semantics was done.

Although the system was designed as to use a prosodic analysis component, currently it does not include such a component. However, to evaluate the effect of using prosodic analysis, we tailored the system in such a way that it can optionally use transplanted prosody from the original speech utterances. This approach was used in the experiments to see the effect of real prosody on the output speech quality.

4. UNIT SELECTION USING VITERBI ALGORITHM

The output of the analysis module is a sequence of phonemes (units) corresponding to the input text, each having energy, pitch, and duration values. We refer to this sequence as the target sequence. The speech corpus had already been processed to build a unit inventory storing the phonemes with the same prosodic features (energy, pitch, duration) and the context information. The unit selection module tries to choose the optimal set of units from the unit inventory that best match the target sequence.

Optimal unit selection algorithm we used is a Viterbi best-path decoding algorithm that is very similar to the one used in CHATR speech synthesis system and is described below [3].

Given a target sequence $t_1^n = (t_1,...,t_n)$, the problem is finding the unit sequence $u_1^n = (u_1,...,u_n)$ that optimizes a cost function of the distance between the two sequences. There are two kinds of cost function in unit selection, namely target cost and concatenation cost. Target cost (unit cost) is an estimate of the cost of using a selected unit in place of the target unit. This cost is a measure of how well the unit from the unit inventory suits the corresponding target unit in the target sequence. This cost can be calculated as a weighted sum of the target sub-costs as follows:

$$C^t(t_i,u_i) = \sum_{j=1}^{P} w_j^t C_j^t(t_i,u_i)$$

where P is the number of target sub-costs and w_j^t are the corresponding weights.

The concatenation (join cost) is an estimate of the cost of concatenating two consecutive units. This cost is a measure of how well two units join together in terms of spectral and prosodic characteristics. The concatenation cost for two units that are adjacent in the unit inventory is zero. Therefore, choosing adjacent units in unit selection is promoted resulting in better speech quality. This cost can be calculated as a weighted sum of the concatenation sub-costs as follows:

$$C^c(u_i,u_{i+1}) = \sum_{j=1}^{Q} w_j^c C_j^c(u_i,u_{i+1})$$

where Q is the number of concatenation sub-costs and w_j^c are the corresponding weights.

The total cost of selecting a unit sequence u_1^n given the target sequence t_1^n is the sum of the target and concatenation costs:

$$C(t_1^n, u_1^n) = \sum_{i=1}^{n} C^t(t_i, u_i) + \sum_{i=1}^{n-1} C^c(u_i, u_{i+1})$$

The unit selection algorithm tries to find the unit sequence u_1^n from the unit inventory that minimizes the total cost.

Since the number of units in unit inventory is very large, we employed some pruning methods to limit the number of units considered. By making use of a window size of three, for a target unit, we select only those units whose left and right three units are identical to those of the target unit. If there exist no such units, the search is repeated with a window size of two and finally with a window size of one.

In calculating the target sub-costs $C_j^t(t_i, u_i)$, we use the context match length, energy, duration and pitch difference between the target and the selected units, and the location of the unit within the syllable, word and sentence. For the concatenation sub-costs $C_j^c(u_i, u_{i+1})$, we use the cepstral distance and the energy, duration and pitch difference between the consecutive units. The cepstral distance at the concatenation points of two consecutive units is an objective measure of the spectral mismatch between these joining units. We use Mel-Frequency Cepstrum Coefficients (MFCCs) for this purpose. We extract the MFCC of the last frame of the first unit and the first frame of the second unit and then use the distance between these two MFCC vectors as the cepstral distance.

5. UNIT CONCATENATION AND WAVEFORM GENERATION

The waveform generation module concatenates the speech waveforms of the selected units. We used a speech representation and waveform generation method based on harmonic sinusoidal coding of speech [6]. Analysis-by-synthesis technique was used for sinusoidal modeling.

The sinusoidal coding encodes the signal with a sum of sinusoids whose frequency, amplitude, and phase are adequate to describe each sinusoid. The harmonic coding is a special case of the sinusoidal coding where the frequencies of the sinusoids are constrained to be multiples of the fundamental frequency.

A perfectly periodic signal can be represented as a sum of sinusoids:

$$x[n] = \sum_{k=0}^{T_0-1} A_k \cos(nk\omega_0 + \phi_k)$$

where T_0 is the fundamental frequency of the signal, $\omega_0 = 2\pi / T_0$, ϕ_k is the phase of the kth harmonics, and A_k is the amplitude of the kth harmonics. For the quasiperiodic speech signals, the same equation can be used to approximate the signal. This approximation can even be used to model the unvoiced sounds. In this case, the fundamental frequency is set to 100 Hz. The error in representing the speech by a harmonic model is estimated as:

$$\varepsilon = \sum_{k=-T_0}^{T_0} \omega^2[k](x[k] - \tilde{x}[k])^2$$

where ω is a windowing function, x is the real speech signal and $\tilde{x}$ is the harmonic model for the speech signal. For parameter estimation of the harmonic coding, we use this function for error minimization criterion. The values for A_k and ϕ_k that minimize the error are calculated by solving the linear set of equations obtained by integrating the error function. Finding model parameters is a least squares problem. We used QR factorization method for solving the set of linear equations to obtain the model parameters.

The correct pitch period estimation is an important part of harmonic coding. The algorithm that we used for pitch estimation is based on the normalized autocorrelation method. The normalized autocorrelation is calculated as:

$$R_n(k) = \frac{\sum_{n=0}^{N-1} x[n]x[n+k]}{\sqrt{\sum_{n=0}^{N-1} x^2[n] \sum_{n=0}^{N-1} x^2[n+k]}}$$

The model parameters are calculated in a pitch-synchronous manner using overlapping windows of two pitch periods. The scalar quantization of model parameters is performed. The unit speech inventory was compressed about three times using quantized model parameters.

The waveform generation using the model parameters for speech waveforms of units is done by taking the inverse FFT of the parameters and then overlap-and-add mechanism is used for smooth concatenation of the units.

6. EXPERIMENTS AND RESULTS

To evaluate the quality of the synthetic voice produced by the developed system, we carried out formal listening tests. The tests were of two types. The first one requires the listeners to rank the voice quality using a Mean Opinion Score (MOS) like scoring. The other test is a diagnostic rhyme test.

The MOS test was carried out by synthesizing a set of 50 sentences. [1] 10 subjects (2 females) were used and they listened the sentences using headphones. The sentences were at 16kHz and 16 bits. We built five different systems and evaluated their quality. The first system uses the original recordings from the test speech corpus that were coded by our harmonic coder and reconstructed (system A). The second system uses the unit selection synthesizer with a speech unit inventory containing about 19 hours of speech recording (system B). The third system uses a speech inventory containing about 3 hours of recording (system C). The last two systems, systems D and E, are the same as systems B and C, respectively, except that the original prosody from the original recordings is used in the unit selection process.

The subjects gave ratings in terms of intelligibility, naturalness, and pleasantness to each sentence. The average MOS scores are shown in descending success rates in Table 1. Figure 2 shows the scores for each category. The differences in system ratings were found to be significant using ANOVA analysis. The analysis yielded an F-value of about 21 whereas the critical F-values are about 3.3 and 5.0 for P=0.01 and P=0.001, respectively.

[1] http://www.cmpe.boun.edu.tr/~gungort/publications/turkishttssamples.htm.

Table 1. Systems and average scores for the MOS test

System	Description	MOS
A	The original recordings with harmonic coding	4.91
B	Speech synthesis using 19 hours of speech	4.20
C	Speech synthesis using 3 hours of speech with original prosody	4.11
D	Speech synthesis using 19 hours of speech with original prosody	4.01
E	Speech synthesis using 3 hours of speech	4.00

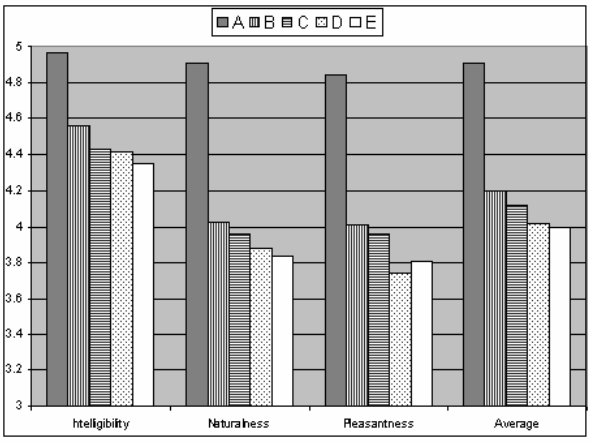

Figure 2. MOS scores with respect to test category

We also conducted an intelligibility test. Diagnostic rhyme test (DRT) uses monosyllabic words that have consonant-vowel-consonant pattern. This test measures the capability of discrimination of the initial consonants for the system evaluated. We constructed a DRT word list for Turkish based on the categories of the DRT word list of English as shown in Table 2.

Table 2. DRT word list for Turkish

Voicing	Nasality	Sustenation	Sibilation	Graveness	Compactness
var far	mal bal	van ban	cent kent	biz diz	türk kürk
ben ten	mat bat	ve be	saç taç	pas tas	fan han
gez kez	naz daz	var bar	sez tez	boz doz	ver yer
bul pul	mil bil	şap çap	jön yön	pek tek	faz haz
din tin	mit bit	vur bur	jel gel	pers ters	dün gün
diz tiz	mor bor	şam çam	sin tin	fon ton	tap kap
zor sor	mut but	şan çan	zan tan	post tost	tuş kuş
zevk sevk	mir bir	fes pes	say tay	put tut	toz koz
zar sar	muz buz	şark çark	zam tam	pak tak	tas kas
zen sen	nam dam	fil pil	zat tat	poz toz	taş kaş
zil sil	nar dar	şal çal	zerk terk	pür tür	tat kat
bay pay	nem dem	şık çık	çal kal	bağ dağ	tel kel
ders ters	nur dur	şok çok	sak tak	bul dul	düz güz
gör kör	nal dal	fas pas	çil kil	bel del	tül kül
vay fay	nil dil	fark park	çim kim	but dut	ton kon
göl çöl	men ben	fiş piş	san tan	fer ter	tork kork

Using the DRT word list for Turkish, we carried out an intelligibility test for our system. The randomly selected words from each pair of the DRT word list were synthesized using the system. The output speech waveforms were played to 10 native Turkish listeners who were then asked to choose which one of the words given in pairs from the DRT list they heard. The test results are shown in Table 3 as the percentage of the number of correct selections for the two systems evaluated.

Table 3. Systems and average scores for the DRT test

System	Description	DRT
B	Speech synthesis using 19 hours of speech	0.95
E	Speech synthesis using 3 hours of speech	0.94

7. CONCLUSIONS

In this paper, a corpus-based concatenative speech synthesis system architecture for Turkish has been proposed and implemented. A pronunciation lexicon for the root words in Turkish has been prepared. A text normalization module and a grapheme-to-phoneme conversion module based on morphological analysis of Turkish have been implemented. Speech corpus has been compressed by a factor of three using a speech model based on the harmonic coding. A DRT word list for Turkish has been constructed to carry out the intelligibility tests. The final system is capable of generating highly intelligible and natural synthetic speech for Turkish and got 4.2 MOS like score and 0.95 DRT correct word discrimination percentage.

REFERENCES

[1] Y. Sagisaka, N. Iwahashi, and K. Mimura, "ATR v-TALK Speech Synthesis System", in. Proc. of the ICSLP, 1992, pp. 483-486.

[2] A. W. Black and P. Taylor, "CHATR: A Generic Speech Synthesis System", in Proc. of the International Conference on Computational Linguistics, 1994, pp. 983-986.

[3] A. Hunt and A. W. Black, "Unit Selection in a Concatenative Speech Synthesis System Using a Large Speech Database", in Proc. of the IEEE on Acoustics and Speech Signal Processing, Munchen, Germany, 1996, pp. 373-376.

[4] M. Beutnagel, A. Conkie, J. Schroeter, Y. Stylianou, and A. Syrdal, "The AT&T Next-Gen TTS system", in Proc. of the Joint Meeting of ASA, EAA, and DAGA, Berlin, Germany, March 1999, pp. 18-24.

[5] A. Conkie, "Robust Unit Selection System for Speech Synthesis", in Proc. of the Joint Meeting of ASA, EAA and DEGA, Berlin, Germany, March 1999.

[6] Y. Stylianou, "Applying the Harmonic Plus Noise Model in Concatenative Speech Synthesis", IEEE Trans. on Speech and Audio Processing, vol. 9, no. 1, pp. 21-29, Jan. 2001.

[7] R. E. Donovan, "Current Status of the IBM Trainable Speech Synthesis System", in Proc. of the 4th ISCA Tutorial and Research on Speech Synthesis, Edinburgh, 2001.

[8] X. Huang, A. Acero, and H. W. Hon, Spoken Language Processing, Prentice Hall PTR, New Jersey, 2001.

[9] http://www.phon.ucl.ac.uk/home/sampa/home.htm.

[10] K. Oflazer and S. Inkelas, "A Finite State Pronunciation Lexicon for Turkish", in Proc. of the EACL Workshop on Finite State Methods in NLP, Budapest, Hungary, April 13-14, 2003.

[11] K. Oflazer, "Two-level Description of Turkish Morphology", Literary and Linguistic Computing, vol. 9, no. 2, 1994.

[12] T. Güngor, Computer Processing of Turkish: Morphological and Lexical Investigation, Ph.D. Thesis, Boğaziçi University, 1995.