



“Syllable” Concatenation for Text to Speech Synthesis for Devnagari Script

Mrs Minakshee patil

Department of E & TC
Sinhgad academy Of Engg,
Pune. India

Dr.R.S.Kawitkar

Department of E & TC
Sinhgad College of Engineering
Pune. India

Abstract— This document gives TTS, an abbreviation for Text to Speech is a system which converts the input text to a corresponding sound output. There are various techniques available for TTS. Out of those techniques, this paper uses a combination of database and word breaking into syllables and if necessary barakhadi. A simple and limited database of words is beauty of this project. A failed database search invokes barakhadi breaking.

Keywords— TTS, wavfile, database, barakhadi, synthesis, joint point, syllables

I. INTRODUCTION

Speech synthesis is the artificial production of human speech usually produced by means of computers. Speech synthesis systems are often called text to speech (TTS) systems in reference to their ability to convert text into speech. In speech synthesis, the input is standard text or a phonetic spelling, and the output is a spoken version of the text.

Speech generation is the process, which allows the transformation of a string of phonetic and prosodic symbols into a synthetic speech signal. The quality of the result is a function of the quality of the string, as well as of the quality of the generation process itself. Text-to-speech synthesis is a research field that has received a lot of attention and resources during the last couple of decades for excellent reasons. One of the most interesting ideas is the fact that a workable TTS system, combined with a workable speech recognition device, would actually be an extremely efficient method for speech coding.

It is necessary to find out what is requested today from a text-to-speech system. Usually two quality criteria are proposed. The first one is intelligibility, which can be measured by taking into account several kinds of units (phonemes, words, phrases). The second one, more difficult to define, is often labeled as pleasantness or naturalness. Actually the concept of naturalness may be related to the concept of realism in the field of image synthesis; the goal is not to reconstitute the reality but to suggest it. Thus, listening to a synthetic voice must allow the listener to attribute this voice to some pseudo-speaker and to perceive some kind of expressivities as well as some indices characterizing the speaking style and the particular situation of elocution. For this purpose the corresponding extra-linguistic information must be supplied to the system.

Most of the present TTS systems produce an acceptable level of intelligibility, but the naturalness dimension, the ability to control expressivities, speech style and pseudo-speaker identity still are poorly mastered. However, users demands vary to a large extent according to the field of application: general public applications such as telephonic information retrieval need maximal realism and naturalness, whereas some applications involving professionals (process or vehicle control) or highly motivated persons (visually impaired, applications in hostile environments) demand intelligibility with the highest priority.

II. BLOCK DIAGRAM

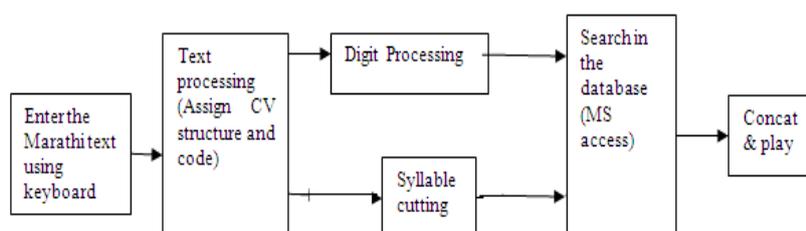


Figure1 Block diagram of Speech Synthesis

III. TEXT ENCODING

The text entered through keyboard is saved in output.doc file. The text encoding stage assigns each character a predefined code. The entire text is converted into a string of code values. Codes are assigned to each character keeping in mind the purpose of the software i.e. speech synthesis. So separate codes For example code of H (full consonant) is 72 while code of H² (half consonant) is 145.. Separate ranges of codes are assigned for consonants and vowels as shown below,

Full Consonants : 72 to 105
Half Consonants : 145 to 178
Vowels : 65 to 71 and 117 to 139

Two ranges of vowels divide the actual consonants and suffixes used for vowels. The range 65 to 71 is given for vowels A Am B B² C D E Eo Amo Am_i A² A: and the other range 117 to 139 is given for all the suffixes used for the vowels ,o p , s , w , y , i , §

Each word is represented as a string of code values separated by "|". The words converted into the code string are given in following table

Sr.No.	Word	Assigned code
1.	MH«Ymas	77 145 97 89 116 97 120
2.	H¥iUgma	145 105 174 85 102 116 170
3.	Amg_mZ	65 116 102 95 116 163
4.	H m _i ñVw^	72 116 123 175 86 138 167

Table1 : code string for different words

The next stage in the text encoding stage is inserting the *rafars*, *kanas*, *anuswars* & *matras* in proper position. The pronunciation of *rafar* is before the character on which it is present whereas the pronunciation of *anuswar* is after the character on which it is present. For example, in the word A&§VZ&©X the pronunciation of *rafar* is immediately after V while pronunciation of *anuswar* is after A& .So the word A&§VZ&©X is actually stored as A& § V © Z&X.

The pronunciation of *anuswar* depends on the character that follows it. So when *anuswar* occurs, it is replaced by an appropriate half consonant depending on the next character in the word. However when *anuswar* comes as the last character of the word it has no nasal sound. Table shows the replacement for some of the characters with suitable examples.

Next Character	Replaced Character	Example
क, ख, ग, घ	ड	अक - अडक
प, फ, ब, भ	फ	गंभीर - गम्भीर
र, ल, उ, ळ	ण	रंज - रणज
द, ध, दध	=	चंद - चन्द
च, छ, ब, भ	ञ	पंच - पञ्च

Table2 : Anuswar Rules

If the character is either ¥ or « or ¢ or ² then the previous consonant is made half while the character is replaced by appropriate full consonant. Some examples are illustrated below.

d¥ is replaced by d² and é
à is replaced by n² and a
öçm is replaced by ö² and `m

IV. DATABASE MAINTAINED

Two databases are maintained viz. Audio database that stores the audio files and Textual database that stores the text files corresponding to audio files in the audio database. The textual database is required to search the required word in the audio database.

Audio Database

WAVE file format is used for recording files. WAVE files are probably the simplest of the common formats for storing audio samples. Unlike MPEG and other compressed formats, WAVE files store samples “in the raw” where no preprocessing is required other than the formatting of the data. A separate sound file is created for each word. Details of .wav file are as below

Audio Format	Pulse Code Modulation
Sampling Rate	11,025 KHz
Bits per sample	8 bits
Channel type	Mono
Audio Format	PCM
Bit Rate	88 kbps

Table3 : Details of recorded .wav file

Textual database

It consists of four entities

- 1) Code string of a word and syllable.
- 2) word and syllable
- 3) Wave filename that contains that word/syllable.
- 4) The from and to positions of the word/syllable within that audio file.

MS Access database is used due to its simplicity & ease of use.

Table shows a format of a table in the textual database. Access database contains only last four columns i.e. code, word, wave file name, from and to position.

WORD	CODE	FILENAME	FROM	TO
चावी	77 116 99 120	ch.wav	132224	137247
चमक	77 95 145	ch.wav	137920	142272
चांगले	77 116 178 74 98 122	ch.wav	142400	147743
चरखा	77 97 73 116	ch.wav	148352	154303

Table4 : Snapshot of textual database(words)

syllable	code	FN (I)	F (I)	T (I)
छर	91 116 170	p.wav	58816	62208
फरु	91 171	p.wav	67072	68671
लख	98 73	l.wav	62208	65984

Table5 : Snapshot of textual database(syllables)

V. DATABASE SEARCH

Database search is carried out in different conditions. They are as follows.

1. Word is present in database : The output from the text encoding stage is a string of code values separated by a “|”. Now this string is searched in the textual database. If match is found it returns the name of the corresponding audio file along with from and to positions. This information is stored in a intermediate file which is later used by the concatenation module. The above method is repeated for the entire text. The output of this module will be a file containing the wave file names along with from and to positions for each word found in the textual database.
2. The word is not present in database : If the word is not present in database, the word is split into syllables. Syllables are maintained into the database same as that of the words. Syllables are formed from the word using CV(consonents- vowel) rules of Marathi language. In the database, code for the syllable, start and end position of it in the word and file name is maintained.

e.g $n\&W\textcircled{c} = n\&a^2 + W$

$$An\&aXe\textcircled{H} = A + n\&a^2 + Xa^2 + eH$$

$$An\&a = A + n\&a$$

And now these syllables are searched into the database.

3. The syllable is not present in database: If the syllable is not present in the database then it is cut into the barakhadi.

VI. DIGIT PROCESSING

Numerical figures are unavoidable part of any text. Presently the algorithm is able to convert numerical figures up to 1, 00,000 into speech. Same algorithm can be applied to increase this range. The numbers from 0 to 100 are recorded and stored in a wave file named digits.wav consecutively. The textual database stores the from and to positions of the digits in the digits.wav file. Words शै , हजार and लाख are recorded each being stored separately. To convert figures in the text into speech, the software first checks whether the separated word is a numerical figure. If the separated word is numerical figure then the Digit Processor comes into the picture.

Working of the Digit Processor is explained as follows:

Words one to ninety nine are recorded and stored successively in file named digits.wav, also words शै , हजार AND लाख are recorded and stored in digits. wav. Also, This corresponds to the audio database for digit processing. The textual database stores the from and to positions for each of the numbers stored in the audio database. The to and from positions for each number are stored in a table named "digits " in textual database. The place of the number is identified and then appropriate suffices like शै , हजार OR लाख are attached. e.g. consider synthesis of 12350.

Place wise digits	Number name	Suffix used	Synthetic word
१२	बारा	हजार	बारा हजार
३	तीन	शै	तीन शै
५०	पन्नास	No suffix	पन्नास

Table5: Digit Processing Example

Algorithm

1. Check if the number is greater than 100000. If yes then continue otherwise follow step 12.
2. Get the quotient of no/100000 in variable quo. Store the remainder in variable rem for further processing.
3. Get the from & to position of the quo by querying the table digits in the textual database.
4. Also Get to and from locations for the word लाख
5. Get the quotient for rem/1000 in quo. Store the new remainder in variable rem for further processing
6. Get the from & to position of the quo by querying the digits table in the database.
7. Also get the to and from locations of the word हजार
8. Get the quotient for rem/100 in quo. Again store the new remainder in variable rem for further processing.
9. Get the from & to position of the quo by querying the digits table in the database.
10. Also get the to and from locations of the word शै
11. Finally get the from & to position of the number stored in the rem variable by querying the digits table in the database.
12. If the number is less than 100000 and greater then 1000 than follow steps 5 to 11.
13. If the number is less than 1000 and greater than 100 than follow steps 8 to 11.
14. If the number is less than 100 than get the from & to position of the number by querying the digits table in the database.
15. Stop.

VII. CONCATENATION AND PLAY

This stage concatenates all the wave files for the different words in the text and makes a new wave file. The program receives input as text document containing the names of all the files, which need to be concatenated in order to generate speech corresponding to the text. The program concatenates the wave files written in the filenms.doc to generate target.wav. The samples from individual wave files are read one at a time and appended to the new target.wav file. The process is repeated till all the wave files in the filenms.doc have been concatenated. The data length for the new file

target.wav is updated at every loop and finally written to target.wav. All the wave files are opened in the binary mode. This is done because the data in these files can be accessed only in binary mode.

VIII. CONCLUSION

The word which is present in the database, is not broken into the syllables due to which joint point is absent and hence it sounds natural. If the word is not present in the database, it breaks into syllables and if syllable is not present then only it is broken into barakhadi. Due to syllable approach presence of joint point reduces and the naturalness of the speech output comparatively increases.

REFERENCES

- [1] O' Brien, D. Monaghan, "Concatenative synthesis based on harmonic model", *IEEE Transactions on Speech and Audio Processing*, Vol.9 [1], Jan 2001, 11-19.
- [2] Hamza, Rashwan and Afifi, "A quantitative method for modeling context in concatenative speech synthesis using large speech database", *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2001.
- [3] Klatt Denis H., Allen Jonathen, Hunnicutt Sharon M., From "Text To Speech: The MITalk System", [Cambridge University Press, London 1987].
- [4] Bureau of Indian Standards [1991], Indian Script Code for Information Interchange IS 13194:1991. New Delhi, India.
- [5] Y.A. El-Imam, "An Unrestricted vocabulary Arabic synthesis system", *IEEE Transactions, Acoustics, Speech and Signal Processing*, Vol.37 [12], 1989, pp1829-1845
- [6] J. Allen, S. Hunnicut, D.Klatt, "From Text To Speech, The MITTALK System", [Cambridge University Press, 1987, 213 pp]