# A RESEARCH BED FOR UNIT SELECTION BASED TEXT TO SPEECH SYNTHESIS

**K Partha Sarathy#, A G Ramakrishnan**

Department of Electrical Engineering, Indian Institute of Science, Bangalore 560012, India
Centre for Development of Telematics#, Bangalore 560100, India
parthu143@gmail.com#, ramkiag@ee.iisc.ernet.in

## ABSTRACT

The paper describes a modular, unit selection based TTS framework, which can be used as a research bed for developing TTS in any new language, as well as studying the effect of changing any parameter during synthesis. Using this framework, TTS has been developed for Tamil. Synthesis database consists of 1027 phonetically rich pre-recorded sentences. This framework has already been tested for Kannada. Our TTS synthesizes intelligible and acceptably natural speech, as supported by high mean opinion scores. The framework is further optimized to suit embedded applications like mobiles and PDAs. We compressed the synthesis speech database with standard speech compression algorithms used in commercial GSM phones and evaluated the quality of the resultant synthesized sentences. Even with a highly compressed database, the synthesized output is perceptually close to that with uncompressed database. Through experiments, we explored the ambiguities in human perception when listening to Tamil phones and syllables uttered in isolation, thus proposing to exploit the misperception to substitute for missing phone contexts in the database. Listening experiments have been conducted on sentences synthesized by deliberately replacing phones with their confused ones.

***Index Terms***— speech synthesis, speech codecs, intelligibility, naturalness, perception

## 1. OVERVIEW OF THE TTS ENGINE

Text-to-Speech (TTS) synthesis involves the production of a speech signal corresponding to the input text. TTS synthesis is inherently multidisciplinary, which includes engineering, multiple areas of linguistics, computer science and acoustics. A TTS engine consists of two major functional blocks: *Natural Language Processing (NLP)*, which is language dependent and *Digital Signal Processing (DSP)*, which is language independent. The NLP block produces a phonetic transcription of the given input text, with prosodic and pause tags. The DSP block transforms the phonetic transcription into speech. Two main classes of TTS systems have gained significance: (i) Synthesis by rule (ii) Concatenative synthesis. Formant and articulatory synthesizers belong to the former category [1]. Concatenation based synthesizer generates speech by concatenating speech segments selected from a large database. It uses a database of recorded speech stored as uncoded waveforms or encoded by a specific coding scheme. Synthesized speech quality is better, if more data is available during training phase. Our TTS framework developed is concatenation based.

## 2. PHONETICALLY RICH DATABASE

Many a times, it is observed that the quality of synthesized speech suffers due to the non-availability of data units matching the target specifications, especially in concatenation-based synthesis. A very careful and systematic data collection will solve nearly half of the synthesis problems. The problem of selecting a set of phonetically rich sentences from a corpus can be mapped to the Set Covering Problem (SCP). The most widely used algorithms for SCP are greedy algorithms [4][5].

### 2.1 CIIL text corpus

The Tamil text corpora used has 30 lakhs words and is from the multilingual text corpus distributed from Central Institute of Indian Languages. The source of this corpus is a large number of printed books, journals, magazines, newspapers and government documents.

### 2.2 Creation of the speech database

A greedy algorithm [5] is used to select 1027 phonetically rich sentences from the text corpus. The selected sentences are recorded from a professional male speaker at a studio in Chennai, Tamilnadu, India, because it is this speaker's voice, which is played out as synthesized speech. The total size of this synthesis speech corpus, sampled at 16 kHz, is 295 MB. Using their phonetic transcription, we segmented the wavefiles using PRAAT[6] software and by trained people to give exact boundaries of phones. Our synthesis database has 5149 unique words. The coverage of the units is really good. The list of phonemes used for segmentation is shown in figure 1.1.

Ü Ý Þ ß à á â ã ä å æ å÷ ç è é
ê ë ì í î ï ð ñ ò ó ô õ ö ÷ ø ù
ú û ü ý

**Figure 1.1 List of Tamil phonemes used**

## 3. SELECTING OPTIMAL UNITS FOR CONCATENATION

High quality TTS systems concatenate speech units carefully selected from a large database of continuous read speech recorded from a professional speaker. Such a database is designed to cover as much of the prosodic and phonetic characteristics of the language as possible. The selection process retrieves from the database, speech segments that best match the target specification, both phonetically and prosodically. The commonly used units for speech synthesis are phonemes, diphones, triphones, syllables, partnemes [7] and words. In our work, if available, a unit bigger than a syllable is selected and concatenated to produce natural sounding synthetic speech. The idea is to have "*units of larger length and few concatenation points*". The naturalness of synthesized speech depends on the unit selection criteria [3].

## 4. SYNTHESIS AND EVALUATION

The performance of any TTS engine is determined by the intelligibility and naturalness of the synthesized speech. Our Tamil TTS is tested for the same. The frame work has been tested for a set of sentences and some sample synthesized sentences are available on IISc website at
http://ragashri.ee.iisc.ernet.in/MILE/index_files/research%20area.html
We use mean opinion score (MOS) to assess the quality of our TTS output. MOS rating is defined as follows: 5 – Excellent; 4 – Good; 3 – Fair; 2 – Poor; 1 – Bad. Ten Tamil sentences, which are distinct from the database sentences, are synthesized. High scores in the range of 4.5-4.8 have been obtained, which made the platform suitable for our research purposes.

## 5. EMBED TTS IN MOBILE USING SPEECH CODING

Theoretically, to synthesize highly natural sounding speech, unit selection based systems need large hardware resources; a much higher order storage and random access memory. Due to the technology advancements in digital memories, many devices with sufficient storage are available. However, mobiles are resource-challenging applications. TTS in a mobile is a value added service, since it is not used continuously. It shall be invoked in mobiles only when necessary. Existing codec in mobiles can be utilized for TTS, when it is integrated into them.

### 5.1 Database compression using speech codecs

Several commercial speech codecs are used to compress our speech database. Some of them include GSM Full rate codec (13Kbps), GSM Enhanced Full rate codec (12.2 Kbps) and GSM Adaptive multi rate codec (4.75-12.2 Kbps) [8]. The lowest and highest compression rates correspond to 13 Kbps and 4.75 Kbps. The size of uncompressed database is 295 Mbytes while the size gets reduced to 17 Mbytes on using high compression. During synthesis, the appropriate compressed segment is selected from the database, decoded and used for concatenation. We synthesized 220 wavefiles using all the compression schemes. The perception experiments evaluated that TTS engine produced a high quality synthetic speech, even with highly compressed database. The complete results of the subjective evaluation are elaborated in [9].

## 6. SUBSTITUTE PHONES – EXPLOIT HUMAN PERCEPTION

Corpus based TTS makes use of a large amount of phonetically rich pre-recorded data in the database. The intention of such a database is to include all possible phonetic contexts of each phone in the language. This is theoretically true but practically it is impossible to capture all the phonetic contexts using any finite database. When a required phonetic context is not available in the database, a simple solution is to automatically note that down and later add a pre-recorded segment corresponding to this context to the database. This, however, increases the database size. In the case of specific phones, we suggest that we can determine a possible phonetic context, similar to the unavailable context, and use it such that the listeners don't distinguish it perceptually. We exploited human perception and found out pairs of phonemes in Tamil language that are generally, perceptually undistinguished. The existence of an 'image phone' is thus justified. Perception experiments along with a phoneme classification experiment supported the above fact. The classification experiments are conducted on our Tamil database containing 1027 sentences.

### 6.1 Perception experiments

Listening experiments are conducted over the telephone to capture the most 'confused' phones in Tamil language. One native of the language calls another and pronounces a list of 152 phones/ syllables (combination of consonant and vowel) in Tamil language in a random order, and the person on the other side writes the phones he/she perceives. Repetition of phones by the speaker is not allowed. Individual phones are chosen to find the exact confusion between phones; if words are used, the listener who has a prior knowledge of the word may write the word correctly even though he may not have perceived properly or the word is not pronounced properly and hence, it would not serve the purpose. The experiment is conducted with 10 pairs of native Tamil people. On an average, 30% of the phones are wrongly identified as other phones. Most of the nasals are wrongly identified as other nasals. Many of the long vowels (*deergha* phones, e.g., 'A' as in the English word 'call') are identified as short vowels (*hrasva* phones, e.g., 'a' as in the English word 'at') and vice versa. The two kinds of /r/ phones in Tamil - /r/ and /R/ - are mis-recognized for each other. The 3 types of /l/ in Tamil - /l/, /L/ and /zh/ - are confused among themselves. Many times, the vowels like /i/, /u/ are identified as combination of a semivowel and vowel - /yi/, /wu/. However, the mis-recognition 'between' different groups of phones like vowels, nasals, fricatives, and glides is relatively less compared to the misrecognition 'within' the groups. The consistently and frequently misrecognized phones are listed in Table 1.1.

**Table 1.1 Phones identified consistently and frequently for one another in Tamil over 10 speaker-listener pairs (/ng/, /ny/, /N/, /n/ are the nasals of /k/, /ch/, /T/, /t/)**

| | | |
|---|---|---|
| Ng – n | A – a | i – yi |
| ny –n | I – i | u – wu |
| N – n | U - u | L – l |
| Ng – ny | S - s | L – zh |
| R – r | | |

### 6.2 Phone classification experiment

After identifying the phones recognized wrongly for other phones, we classified the phones using Maximum Likelihood (ML) classifier. MFCCs are used as spectral features in the experiment. Two types of classifications were performed: frame level and phone level. In the former case, a 10 ms frame (a 12-dimensional acoustic vector) is classified as one of the 48 Tamil phone classes using the ML classifier. In the latter case, the mean of all the MFCC vectors belonging to one phone is taken and classified using ML classifier.

### 6.3 Results of phone replacement

Phones are classified using both full and diagonal covariance matrices. The classification accuracy obtained with full covariance matrix is better than that obtained with the use of diagonal covariance matrix. The experiments are carried out for different sizes of training and test data and the phones that are misclassified are noted down. The results of the experiment are presented in Table 1.2. BCCA (broad class classification accuracy) is the accuracy of correctly classifying a phone to its major category. For Example, if a vowel is identified as a vowel, a nasal is identified as a nasal and so on, the classification is considered to be accurate. The fifth column of Table 1.2 is the accuracy of classifying a phone to its true class. Both of them are found to increase with the size of training data.

**Table 1.2 Phone classification results on 100 sentences from MILE Tamil corpus (full covariance matrix).**

| Expt no | Size of training data | Avg number of feature vectors per class | BCCA | Accuracy |
|---|---|---|---|---|
| 1 | 100 | 6295 | 61% | 47% |
| 2 | 200 | 6938 | 65% | 49% |
| 3 | 400 | 8648 | 72% | 53% |
| 4 | 700 | 10603 | 74% | 53% |

### 6.4 Confusion matrix

A confusion matrix of the Tamil phones is shown in Table 1.3 mainly for significant mismatches. The classification accuracy of the phone /a/ is higher than that of the other phones. Consistently, for all the cases, 25% of the 'a' phones are classified to 'A'. 72% of the /I/ phones are classified as /i/. This is not so prominent with the other vowels. So, if a *deergha* syllable ([consonant + A/I] or [A/I + consonant]) is not available in the corpus in a particular context, it may probably be replaced with the *hrasva* syllable ([consonant + a/i] or [a/i + consonant]). This is a major finding. The confusion between /u/ and /U/ pairs is frequent in the listening tests but not so significant in the classification test. The following are the results from the experiment, where the size of the training data is 700 and that of the test data is 400. 40.9% of /ae/s are classified to /i/ while only 29.8% of /ae/s are correctly classified to /ae/ class. 9% of /ae/s are classified to /yl/ (genitive of /y/). 38% of /yl/s are classified to /i/. 11.4% of /yl/s are classified to /ae/. Thus, there is more misclassification among the three phone classes - /i/, /yl/ and /ae/.

### 6.5 Exploiting the misperception of the phones

Blind listening tests are conducted with four native Tamil people. They are asked to listen to a set of 11 synthesized sentences, which are generated by our Tamil TTS system. The same 11 sentences are also synthesized with certain phones in some words replaced by the corresponding confused phones found. Many words had a single phone replacement and some of them also had 2 to 3 phone replacements. The listeners are asked to write the synthetic sentences of both the sets separately. The results are checked to find the validity of the phone replacement. All the listeners recognize 75% of the words having replaced phone(s) as the regular words. They could get the original word, even though some of the phones had been replaced by other phones in those words. 3 listeners did not notice a change in 50% of the remaining 25% phone replaced words. The replacement of a vowel by consonant-vowel combination at the beginning of the words (e.g., i-yi) and the replacement of phones like nasals and glides with their corresponding confused phones (e.g., /m/-/n/ and /l/-/zh/ respectively) at the end of words worked favorably. Interchanging of the nasals /n/ and /N/ worked at all places. Interchanging /m/ and /n/ at the beginning of the words also worked nicely. Replacement of /l/, /L/ and /zh/ among themselves worked well always. Replacement of /r/ with /R/ was good to some extent. Deergha – Hrasva replacement works well at all places, since the listener who has a prior knowledge of the word, gets the word right, even if there is a lengthening of some vowels.

## 7. CONCLUSION

Our Tamil TTS framework is concatenation based, where units are selected from the phonetically rich database based on spectral matching. A method is proposed for embedding TTS in commercial mobiles. Studies have been conducted to identify the Tamil phones that get confused in human perception. Studies have been conducted to explore the idea that, during synthesis, phones found missing in the database can possibly be substituted with their corresponding *confused phones* to produce intelligible speech, possibly without the need to increase the size of the synthesis database.

## 8. REFERENCES

[1] Dennis H. Klatt, "Review of Text to speech conversion for English", *Journal of Acoustic Society of America,* Vol. 82 No:3, September 1987, pp.210-252.

[2] Ananthapadmanabha TV "Articulatory and Formant Synthesizers" Lecture material, Winter School Speech and Audio Processing, Jan 2-5, 2008, IIT Madras.

[3] A Black and N Campbell, "Optimizing selection of units from speech databases for concatenative synthesis", In *Proc Eurospeech, pp.* 581-584, 1995.

[4] Jan P.H Van Santen and Adam L. Buchsbaum, "Methods for optimal text selection," Proc *Eurospeech'97,* Rhodes, Greece, 1997, pp. 553-556.

[5] Ronald L Rivest, Thomas H Cormen, Charles E Leirson, Introduction to Algorithms, PHI, Delhi, 2000.

[6] PRAAT : A tool for phonetic analyses and sound manipulations by Boersma and Weenink, 1992-2001. www.praat.org

[7] Mukhopadhyay A, Chakraborty S, Choudhury M, Lahiri A, Dey S and Basu A,(2005) "Shruthi: an embedded text-to-speech system for Indian languages", *Software, IEEE Proceedings* – Vol:153, Issue:2, pp. 75-79.

[8] K.Parthasarathy, A.G. Ramakrishnan "Text To Speech synthesis system for mobile applications", Proc WISP-2007, IIT-Guwahati, India.

[9] K.Parthasarathy, "A research bed for unit selection based text to speech synthesis system", MS Thesis, Dept. of Electrical Engineering, IISc, Bangalore, August 2008.

**Table 1.3 Tamil phones perceived with maximum confusion.**

| | True Class | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| Assigned Class | | a | A | i | I | ae | l | ll | yl |
| | a | 3164 | 166 | 180 | 3 | 65 | 6 | 33 | 74 |
| | A | 1112 | 1461 | 0 | 0 | 0 | 4 | 2 | 0 |
| | i | 228 | 0 | 1962 | 110 | 419 | 2 | 6 | 407 |
| | I | 1 | 0 | 9 | 7 | 1 | 0 | 0 | 1 |
| | ae | 112 | 0 | 220 | 11 | 305 | 0 | 0 | 122 |
| | l | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | ll | 0 | 0 | 0 | 0 | 0 | 0 | 12 | 0 |
| | yl | 61 | 0 | 130 | 7 | 92 | 0 | 0 | 378 |
| | Total | 5788 | 1633 | 2909 | 148 | 1023 | 83 | 369 | 1069 |