An Approach to Building Language-Independent Text-to-Speech Synthesis for Indian Languages

Anusha Prakash*, M Ramasubba Reddy*, T Nagarajan† and Hema A Murthy‡
*Department of Applied Mechanics, Indian Institute of Technology Madras, India
†Speech Lab, SSN College of Engineering, Chennai, India
‡Department of Computer Science and Engineering, Indian Institute of Technology Madras, India
Email: hema@cse.iitm.ac.in

***Abstract*—A popular speech synthesis method is the HMM based speech synthesis method. Given the phone set and question set for a language, HMM based synthesis systems are built. Although robotic in quality the systems are intelligible. In this paper, we propose a common framework for Indian languages with a common phone set and a common question set. Owing to this architecture it is possible to borrow independent monophone models across languages. Degradation MOS and word error rate scores are comparable to systems built in the conventional language-specific manner, indicating that system building can be made language-independent without much degradation in the quality of synthesised speech.**

I. Introduction

State-of-the-art text to speech synthesis (TTS) systems are unit selection (USS) and HMM based speech synthesis systems (HTS). In unit selection speech synthesis system [1], the original waveforms are split into units like phones, diphones, syllables, etc and stored in the database during training. During synthesis phase, the waveform units corresponding to the test sentence are chosen from the database based on concatenation criteria and synthesised. Inspite of a careful choice of concatenation criteria USS still suffers from discontinuity at joins. In HTS, context-dependent pentaphone models are built during training [2]. Although the synthesised output is muffled due to averaging, the speech waveform is quite intelligible. The footprint size is also quite small. This makes HTS quite attractive for use in mobile phones, tablets, etc.

Considering the number of languages in India, building systems for each of them is quite tedious. For building a TTS system from scratch for a new language, one needs to have a thorough understanding of the phonotactics of the sounds in the language, letter to sound (LTS) rules, etc. This requires help from linguists of that language and the process becomes complex and time-consuming. Thus building new systems from existing systems definitely has more appeal.

In the field of speech recognition, borrowing of acoustic models across languages have been quite successful. In [3], [4], [5], the phoneme mapping from source language(s) to target language is obtained through data-driven or knowledge-based approaches. Next, context-independent acoustic models are copied to the target language. With some adaptation techniques, waveforms of target language are segmented into phonemes. In some instances, context-dependent acoustic models are also borrowed across languages [6].

Unlike [7], where phoneme mapping is obtained automatically, common phone set for Indian languages is used for mapping [8]. Context-independent monophone models are copied to the target language from the source language. Viterbi training is performed to obtain annotated training data of target language. A common question set designed from the common phone set is used for decision tree based clustering [8]. Waveforms with corresponding text along with letter to sound rules are the only language dependent requirements. Additionally, an experiment using a common parser is also performed. The goal is to make system building as language-independent as possible, so that TTS systems can be built with ease even with no knowledge about the language. This is the main work presented in the paper.

The paper is organised as follows. The motivation behind the work is explained in Section II. Section III gives a brief overview of HMM-based speech synthesis system. In Section IV, experiments and results are detailed. The results are discussed in Section V. The work is concluded and future work is discussed in Section VI.

II. Motivation

There are 1652 languages in India. Building a TTS system for each of them is time-consuming and exhausting. Thus a more generic approach towards system building is required. A common framework is first designed, using which language-specific systems are then built.

Most of the Indian languages can be classified as Indo-Aryan or Dravidian. These classes of languages share some common features. Owing to significant mixing of the Dravidian and Aryan races, there has also been lot of borrowing of sounds

978-1-4799-2361-8/14/$31.00

from the languages. Exploiting this fact, a common phone set is designed. A common question set is then derived from the common phone set. The common phone set and common question set of [8] are briefly described.

A. Common Phone Set

All the sounds or phonemes of 13 Indian languages are listed. Similar sounds across different languages are mapped together and denoted by a single label. These labels are represented using the Roman alphabet set. Since the number of sounds in a language exceeds the number of roman characters, certain suffixes are used. Suffix x is used to denote retroflex place of articulation, for eg: /d/ vs /dx/. For aspiration, suffix h is used, for, eg: /g/ vs /gh/. Certain sounds are language-specific. They are represented by separate labels. /zh/ is predominant in Tamil and Malayalam. Certain phonemes in Hindi like /kq/, /khq/ have been added to account for some pronunciations in foreign languages. Marathi has both dental and palatal affricates (/c/, /cx/, /j/, /jx/) compared to other languages which have only palatal affricates (/c/, /j/). Some languages have both phoneme and grapheme representations; this is to ensure that the native script is largely recoverable from the transliterated text. The International Phonetic Alphabet (IPA) symbols are used as references. A part of the common phone set is shown in Fig 1. [1]

Label	IPA	Hindi	Marathi	Bengali		Tamil	Malayalam	Telugu
				P	G			
a	/a/	अ	अ	-	-	அ	അ	అ
ax	/ɔ/	-	ऑ	অ	অ	-	-	-
aa	/a: /	आ	आ	আ	আ	ஆ	ആ	ఆ
i	/ɪ/,/i/	इ	इ	ই,ঈ	ই	இ	ഇ	ఇ
ii	/i: /	ई	ई	-	ঈ	ஈ	ഈ	ఈ
rq	-	ऋ,ॠ	ऋ,ॠ	ঋ,ৠ	ঋ,ৠ	-	ഋ	ఋు,ఋూ
e	/e/	-	-	এ	এ	எ	എ	ఎ
ee	/e: /	ए	ए,ऐ	-	-	ஏ	ഏ	ఏ
ei	/ɛ: /	ऐ	-	-	-	-	-	-
ai	/aI/	-	ऐ	-	-	ஐ	ഐ	ఐ
oi	/oj/	-	-	ঐ	ঐ	-	-	-
c	/ʧ/	च	च	চ	চ	ச	ച	చ
ch	/ʧʰ /	छ	छ	ছ	ছ	-	ഛ	ఛ
cx	/ʧ̲/	-	च	-	-	-	-	-
j	/ʤ/	ज	ज	জ,য	জ	ஜ	ജ	జ
jh	/ʤʰ /	झ	झ	ঝ	ঝ	-	ഝ	ఝ
jx	/ʤ̲/	-	ज	-	-	-	-	-
lx	/ɭ/	-	ळ,ळ	-	-	ள	ള	ళ
z	/z /	ज़	ज़	জ়	-	-	-	-
dxq	/ɽ/	ड़	ड़	ড়	ড়	-	ട	-
dxhq	/ɽʰ /	ढ़	ढ़	ঢ়	ঢ়	-	-	-
f	/f/	फ़	फ़	-	ফ়	◌ॅ◌	-	-
rx	/ɾ /	-	-	-	-	ற	റ	ఱ
zh	/ɻ /	-	-	-	-	ழ	ഴ	-
q		◌ं	◌ं	-	◌ং	-	◌ം	◌ం
hq		◌ः	◌ः	-	◌ঃ	-	◌ഃ	◌ః

Fig. 1: Common Phone Set

Around 10 vowels and 33 consonants are present in all languages, except Tamil which has only 26 consonants. Context independent monophone models are built for each of them. They are then used to time align the given data phonetically.

B. Common Question Set

Full context pentaphone is the basic unit used in HTS. For each language, we have approximately 50 phonemes. That will be $(50)^5$ pentaphone models. Additional context information such as position of phoneme in the syllable, position of the syllable in the word, position of the word in the phrase, number of syllables in the phrase, etc are considered. The number of combinations is quite big, and many instances are unavailable in the training data.

A decision tree based state-clustering technique is used to overcome this problem [9]. The question set contains a set of questions that has a yes/no answer for clustering. If the likelihood increases, then the node is split into two. This continues till the likelihood is less than some threshold.

The questions are based on characteristics of the sounds such as vowels, consonants, stop consonants, nasals, front vowels, back consonants, continuents, fricatives, affricates, etc. This requires knowledge of the acoustic-phonetic characteristics of all the sounds in a language. Using the common question set makes the task easier. The common question set is a list of such questions across 6 Indian languages (Hindi, Marathi, Bengali, Tamil, Telugu, Malayalam). An advantage is the structure of scripts of Indian languages according to the place and manner of articulation. 60 questions relevant to Indian languages have been included from the English question set. One Indo-Aryan (Hindi) and one Dravidian (Tamil) language have been chosen as starting points. Most of the sounds have been classified based on studies in [10] on the phonotactics of Hindi characters. These two languages cover most of the phonemes. Additional phonemes from the remaining 4 languages are added to the relevant classifications. An additional phoneme of any new language is included along with a similar phoneme in the common question set. Experiments in [8] have shown that there is not significant degradation in quality by using the common question set.

III. HMM-BASED SPEECH SYNTHESIS SYSTEM

The process of system building can be divided into two phases- training and synthesis. This is illustrated in Fig 2.[2] In the training phase, spectral, excitation (log F0) and duration models are built from the waveforms. Spectral features are mel-generalized cepstral coefficients and their dynamic features

[1] The complete common phone set can be found at http://lantana.tenet.res.in/ilsl12.pdf or http://www.iitm.ac.in/donlab/ilsl12.pdf

[2] This figure has been redrawn from [2]

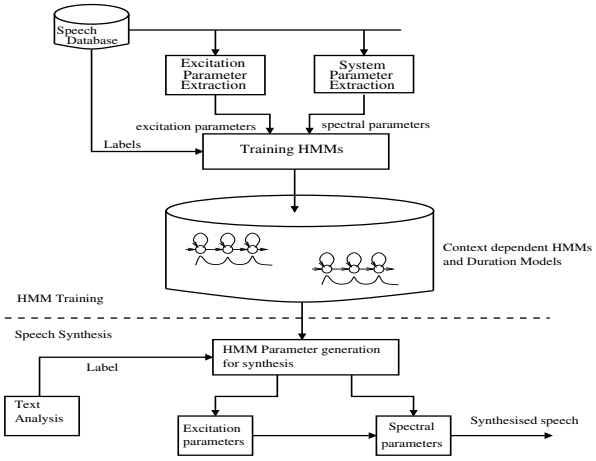

Fig. 2: Training and Synthesis Phases of HMM-based Speech Synthesis

in a single stream. log F0 and its dynamic features model both voiced and un-voiced regions, hence they are multi-space probability distributions in three different streams. The model parameters are estimated based on the maximum likelihood criterion:

$$\hat{\lambda} = arg \max_{\lambda} p(O|W, \lambda) \quad (1)$$

where λ represents the model parameters, O is the training data and W is the transcription corresponding to the training data. Context-independent monophone HMMs are built and re-estimated. Next, context-dependent monophone models are re-estimated. Using a decision tree based clustering, context-dependent pentaphone models are built. State duration probability density functions are derived for every context-dependent model.

In the synthesis phase, context-dependent label sequence is obtained from the test sentence. Corresponding context-dependent models are then selected and concatenated to form the utterance HMM. State durations are determined from state duration probability density functions. State sequence is then obtained from the state durations. Spectral and excitation parameters are generated such that the output probability is maximized

$$\hat{o} = arg \max_{o} p(o|w, \hat{\lambda}) \quad (2)$$

where o represents speech parameters and w is the transcription of the test sentence. During the generation of speech parameters, dynamic features are also taken into account. Using a synthesis filter, speech waveform is then synthesised.

For the target language, context-independent monophone models are borrowed from the source language. The phoneme mapping is obtained from the common phone set. If a new phoneme is present in the target language, it is mapped to a similar phoneme in the source language, and the corresponding monophone model is borrowed.

IV. Experiments and Results

HTS systems are built for 5 Indian languages - Tamil, Malayalam, Telugu, Marathi and Bengali. Text has been selected from different domains such as children's stories, news, etc. Speech data for each language has been collected from a single female speaker recorded at 16 kHz, 16 bits per sample in studio environment.

The conventional manner of system building for any new language involves manual annotation of some amount of data and the design of LTS rules and language-specific question set. But by using a common framework to build upon, the amount of manual effort is reduced tremendously. Using the common phone set and common question set definitely makes the task easier [8]. But by considering context-independent monophone models of a similar language (source language) as initial models for the new language (target language), one can do without manual annotation altogether. Performing forced viterbi alignment on the target language, segmentation of the target language data is achieved.

Hindi and Tamil are chosen as the source languages for other Aryan and Dravidian languages, respectively. For each language, few minutes of data is chosen such that all the phonemes of that language are covered and data is manually annotated at the phoneme level. The phonetic transcription is obtained from a language specific parser. A 5-state, single mixture HMM model is built for each phoneme. These models are then used to time-align the entire database phonetically. 3 hours of the source language data are used. After many iterations, context-independent monophone models are finally obtained which are then used in target languages.

HTS version 2.2 is used to build voices. Experiments are performed by considering different source and target languages. For evaluating the naturalness of the synthesised speech, degradation mean opinion score (DMOS) test is conducted [8], [11]. Word error rate (WER) test is conducted in order to get a measure of the intelligibility of the system. In WER test, evaluators have to transcribe semantically unpredictable sentences [12], [13]. About 8-10 subjects per language are used for evaluation. Results are shown in Table 1. An experiment with dissimilar languages as source and target is also conducted. Two Tamil voices are built- one by manually annotating few minutes of Tamil data and the other by using Hindi models for annotation. The phonetic segmentation of the Tamil word /kuruwin/ is shown in Fig 3. Using Tamil and Hindi monophone models as initial models, the segmentation is quite similar.

For the sake of completeness, an attempt is made to design a common set of LTS rules or parser. The grapheme to phoneme mapping is almost one-to-one in most Indian Languages [14], expect in languages like Hindi where the conversion is more complex involving schwa deletion. A common parser

TABLE I: Degradation MOS (DMOS) and Word error rate (WER) scores

Target Language	Marathi	Bengali	Tamil	Tamil	Telugu	Malayalam
Source Language	Hindi	Hindi	Tamil	Hindi	Tamil	Tamil
Numbers of hours of target language	3	2	3	3	3	3
DMOS	2.79	2.50	2.97	2.53	2.63	2.88
WER	3.48%	15.06%	6.61%	5.16%	16.41%	3.13%

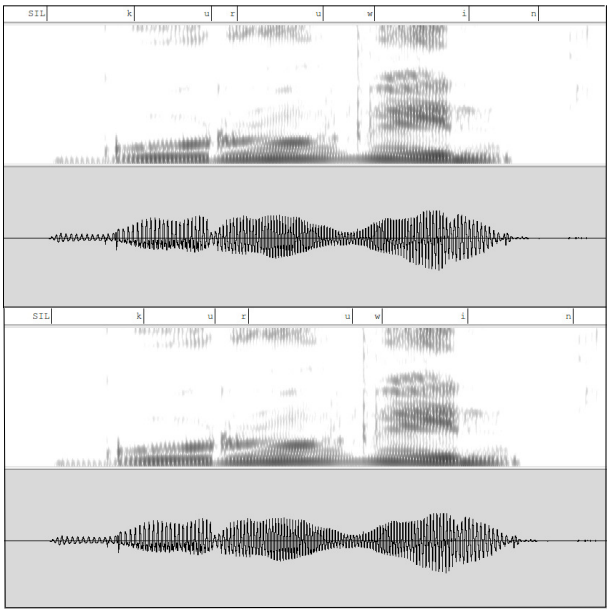

Fig. 3: Phonetic segmentation of /kuruwin/ using Tamil models (top) and Hindi models (below)

is designed for Tamil and Telugu. The common parser has graphemes mapped directly to the phonemes without inclusion of language-specific rules. Informal listening tests indicate that the synthesis quality is quite good.

V. Discussions

An average DMOS of 2.67 and WER of 8.64% is obtained for systems built by cross-language borrowing of context-independent monophone models and using the common phone set and common question set. The DMOS and WER scores for Tamil as both source and target language are from [8]. The scores for Tamil voices built from Tamil models and Hindi models are almost comparable, given that they are dissimilar languages. This reiterates the fact about inter-mixing of languages in the Indian scenario, and the possibility of building one generic system. Thus, the only language-specific requirements for building a TTS system are:

- Text in the native script or in the common phone set transliteration
- Recorded speech data corresponding to the text
- Letter to sound rules or parser that outputs phonetic transcription in common phone set transliteration
- Phoneme list of the language

The system can then be easily built language-independently. If common parsers can be designed for groups of languages, the process can be further automated.

VI. Conclusion and future work

This paper explores an approach to building language-independent text to speech systems for Indian Languages. The use of common phone set, common question set and borrowing context-independent monophone models across languages makes the procedure easier and less time-consuming, without compromising the synthesised speech quality. Systems can be built without even knowing the language. This is especially quite beneficial in the Indian scenario.

This work can also be extended to build TTS systems for under-sourced languages. For languages that have no written script, the common phone set transliteration can be used to build systems.

Acknowledgment

The authors would like to thank the Department of Information Technology, Ministry of Communication and Technology, Government of India for funding the project “Development of Text-to-Speech Synthesis for Indian Languages Phase II” (Ref. no. 11(7)/2011- HCC(TDIL)). The authors would like to thank Gandhi A. N, Kasthuri G. R, Musfir, Jeena Prakash, Lakshmi Priya, Abhijit Pradhan and Asha Talambedu for their help in conducting DMOS and WER tests.

References

[1] A. J. Hunt and A. W. Black, “Unit selection in a concatenative speech synthesis system using a large speech database,” in *Acoustics, Speech, and Signal Processing, 1996 (ICASSP-96)*, vol. 1, 1996, pp. 373–376.

[2] H. Zen, K. Tokuda, and A. W. Black, “Statistical parametric speech synthesis,” *Speech Communication*, vol. 51, no. 3, pp. 1039–1064, November 2009.

[3] A. Beyerlein, W. Byrne, J. M. Huerta, S. Khudanpur, B. Marthi, J. Morgan, N. Peterek, J. Picone, and W. Wang, “Towards language independent acoustic modeling,” in *Proceeding on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 2, 2000, pp. 1029–1032.

[4] R. Bayeh, S. Lin, G. Chollet, and C. Mokbel, “Towards multilingual speech recognition using data driven source/target acoustical units association,” in *Acoustics, Speech, and Signal Processing, 2004. Proceedings ICASSP ’04*, vol. 1, 2004, pp. I–521–4.

[5] V. B. Le and L. Besacier, “First steps in fast acoustic modeling for a new target language: Application to Vietnamese,” in *Acoustics, Speech, and Signal Processing, 2005. Proceedings ICASSP ’05*, vol. 1, 2005, pp. 821–824.

[6] V. B. Le, L. Besacier, and T. Schultz, "Acoustic-phonetic unit similarities for context dependent acoustic model portability," in *Acoustics, Speech and Signal Processing, 2006. Proceedings ICASSP '06*, vol. 1, 2006, pp. I–I.

[7] J. J. Sooful and J. C. Botha, "An acoustic distance measure for automatic cross-language phoneme mapping," in *Pattern Recognition Association of South Africa (PRASA'01)*, November 2001, pp. 99–102.

[8] R. B, S. L. Christina, G. A. Rachel, S. Solomi V, M. K. Nandwana, A. Prakash, A. S. S, R. Krishnan, S. K. Prahalad, K. Samudravijaya, P. Vijayalakshmi, T. Nagarajan, and H. Murthy, "A common attribute based unified hts framework for speech synthesis in Indian languages," in *8th ISCA Workshop on Speech Synthesis*, Barcelona, Spain, August 2013, pp. 311–316.

[9] S. Young, G. Evermann, M. Gales, T. Hain, D. Kershaw, X. A. Liu, G. Moore, J. Odell, D. Ollason, D. P. V. Valtchev, and P. Woodland, *The HTK Book (for HTK Version 3.4)*. Cambridge University Engineering Department, 2002.

[10] P. Eswar, "A rule based approach for spotting characters from continuous speech in Indian languages," PhD Dissertation, Indian Institute of Technology, Department of Computer Science and Engg., Madras, India, 1991.

[11] M. Viswanathan and M. Viswanathan, "Measuring speech quality for text-to-speech systems: development and assessment of a modified mean opinion score (mos) scale," in *Computer, Speech and Language*, vol. 19, 2005, pp. 55–83.

[12] C. Benoit, M. Grice, and V. Hazan, "The SUS test: A method for the assessment of text-tospeech synthesis intelligibility using semantically unpredictable sentences," in *Speech Communication*, vol. 18, no. 4, 1996, pp. 381–392.

[13] C. Benoit, "An intelligibility test using semantically unpredictable sentences: towards the quantification of linguistic complexity," in *Speech Communication*, vol. 9, no. 4, 1990, pp. 293–304.

[14] M. et al, "Building unit selection speech synthesis in Indian languages: An initiative by an indian consortium," in *Proceedings of Oriental COCOSDA*, Kathmandu, Nepal, 2010.