Synthesizing Audio for Hindi WordNet

Diptesh Kanojia[†,♣,⋆], **Preethi Jyothi**[†], **Pushpak Bhattacharyya**[†]
[†]Indian Institute of Technology Bombay, India
[♣]IITB-Monash Research Academy, India
[⋆]Monash University, Australia
[†]{diptesh, pjyothi, pb}@cse.iitb.ac.in

Abstract

In this paper, we describe our work on the creation of a voice model using a speech synthesis system for the Hindi Language. We use pre-existing "voices", use publicly available speech corpora to create a "voice" using the Festival Speech Synthesis System (Black, 1997).

Our contribution is two-fold: **(1)** We scrutinize multiple speech synthesis systems and provide an extensive report on the currently available state-of-the-art systems. We also develop voices using the existing implementations of the aforementioned systems, and **(2)** We use these voices to generate sample audios for randomly chosen words; manually evaluate the audio generated, and produce audio for all WordNet words using the winner voice model. We also produce audios for the Hindi WordNet Glosses and Example sentences.

We describe our efforts to use pre-existing implementations for WaveNet - a model to generate raw audio using neural nets (Oord et al., 2016) and generate speech for Hindi. Our lexicographers perform a manual evaluation of the audio generated using multiple voices. A qualitative and quantitative analysis reveals that the voice model generated by us performs the best with an accuracy of 0.44.

1 Introduction

WordNets have proven to be a rich lexical resource for many NLP sub-tasks such as Machine Translation (MT) and Cross-Lingual Information retrieval (Knight and Luk, 1994; Richardson and Smeaton, 1995). They are lexical structures composed of synsets and semantic relations (Fellbaum, 1998). Such a lexical knowledge base is at the heart of an intelligent information processing system for Natural Language Processing and Understanding. The first WordNet was built in English at Princeton University[1]. Then, followed the WordNets for European Languages[2] (Vossen, 1998), and then IndoWordNet[3] (Bhattacharyya, 2010).

IndoWordNet consists of 18 Indian Languages with an average of 27000+ synsets for all the languages and 40000+ for the Hindi Language. It uses Hindi WordNet[4] (Narayan et al., 2002) as a pivot to link all these languages and contains more than 25000 linkages to the Princeton WordNet. Cognitive theories of multimedia learning (Mayer, 2002) indicate that audio cues are effective aids in a learning scenario, and also help in retaining the material learned (Bajaj et al., 2015).

"Our goal is to enrich the semantic lexicon of Hindi WordNet by augmenting it with word audios generated automatically using a speech synthesis voice model."

Manually recording pronunciations for all the words is a tedious task. These recording efforts could be minimized by using text-to-speech (TTS) systems to automatically synthesize speech for all the words. However, one cannot be sure about the quality of these synthesized clips. We build multiple TTS systems and systematically analyze the quality of the resulting synthesized clips, with the help of

[1] http://wordnet.princeton.edu
[2] http://www.illc.uva.nl/EuroWordNet/
[3] http://www.cfilt.iitb.ac.in/indowordnet/
[4] http://www.cfilt.iitb.ac.in/wordnet/webhwn/

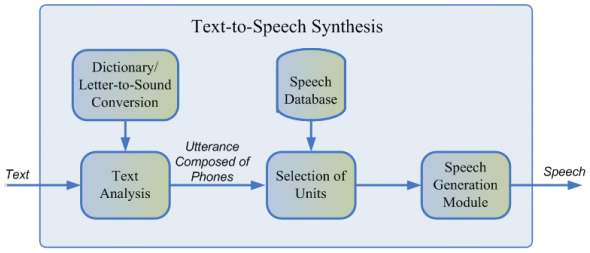

Figure 1: A Unit selection based Concatenative Speech Synthesis System

lexicographers. We envision that this addition to Hindi WordNet will further its use in the education domain, for students and language enthusiasts alike.

1.1 Speech Synthesis: An Introduction

There are four basic approaches to synthesizing speech: 1) waveform concatenation, 2) articulatory synthesis, 3) formant synthesis, and 4) concatenative synthesis. Concatenative synthesis produces a very natural-sounding synthesized version of the utterances. There can be glitches in the output owing to the nature of automatic segmentation of the waveforms, but the speech produced sounds natural indeed. Apart from the first method *i.e.* waveform concatenation, all approaches to speech synthesis are based on the source-filter model. The synthesis method can be broken down into two components, consisting of a model of the source (models of periodic vibration and models of noise supra-glottal sources) and a model of the vocal tract transfer function. In articulatory synthesis, computational models of the articulators are constructed that allow the system to simulate various configurations that human speech organs can attain during speech production. Acoustic-phonetic theory is used to compute the transfer function for vocal tract shape. In formant synthesis, formant transitions across consonants and vowels must be modeled closely. These transitions are most important in identifying the consonant. Designing these set of rules is still a difficult task. The simplest approach to synthesis bypasses most of the problems since it involves taking real recorded/coded speech, cutting it into segments, and concatenating these segments back together during synthesis. It is called concatenative synthesis.

2 Related Work

A significant amount of work has been done in the area of Speech Synthesis or Text-to-Speech conversion for English, Japanese, Chinese, Russian (Takano et al., 2001; Zen et al., 2007; Zen et al., 2009; Wang et al., 2000; Sproat, 1996). Text-to-Speech conversion systems for Indian Languages have also emerged in the recent past (Patil et al., 2013). Although these systems, which are already available, do not produce the most "natural" sounding output, but they are usable to an extent. Manual evaluations of the speech synthesis systems built for the Hindi Language show that there is still a need for better text processing and additional phonetic coverage (Kishore et al., 2003; Raj et al., 2007). Bengu et al. (2002) create an online context sensitive dictionary using Princeton WordNet and implement a Java based speech interface for the Text-to-Speech (TTS) engine. Kanojia et al. (2016) automatically collect images for IndoWordNet and augment them to the web interface, but due to the lack of tagged images openly available for use, they do not collect enough images. To the best of our knowledge, there has been no other work specifically in the direction of synthesizing audio for WordNet words or Synthesizing audio for Indian Language WordNets.

3 Our Approach

Among the three main sub-types of concatenative synthesis, we choose to perform unit selection synthesis and build cluster units of the speech data recorded by a human voice. We use the Festival system to create a synthetic voice for Hindi. We followed the documentation of the Festival Framework along with FestVox[5] implementation to train a voice on Hindi Speech Corpora provided by the IndicTTS Consortium[6] for research purposes. Figure 1 displays a generic speech synthesis system which uses the concatenative synthesis or unit selection corpus-based speech synthesis to generate speech given an input text.

[5] http://festvox.org/

[6] 4https://www.iitm.ac.in/donlab/tts/index.php

3.1 Dataset

We use the Female Voice - Hindi and Female Voice - English dataset provided by the IndicTTS forum to train our system. The dataset is publicly available for the purpose of research. We download the complete dataset i.e. 7.22 hours of Audio with English and 5.18 hours of monolingual audio. We also download the dictionary provided on the website for providing it to the synthesis system as input. We use a total of 2318 Female Hindi sentence utterances downloaded from IndicTTS consortium, and 1378 word audios manually recorded by us to train the voice model.

3.2 Architecture and Methodology

While training input to the system is a corresponding speech-text parallel corpus, where a WAV file containing audio is aligned to its corresponding text using an ID, a textual unit such as a word or a phrase is given as an input in the testing phase. The output is an audio waveform stored in the WAV format.

The system needs a syllable dictionary for a letter to sound conversion and We generate one which contains unique words and in parallel has corresponding syllabification of the word with the beginning and ending clearly marked. For *e.g.*, The Hindi word "*kamaane*" which means "To Earn" would be represented as:

("कमाने" nil (("क_beg") 0) (("मा_mid") 1) (("ने_end") 0))),

0 for lower stress, and 1 for the high stress.

Such a system also requires the utterances composed of phones including a considerable amount[7] of recorded speech.

Both the requirements above can also be generated programmatically from a given text corpora which corresponds to a speech database/corpora. Although, the recorded speech needs to be in parallel correspondence to the recorded audio files (usually in a WAV audio file format - 16KHz, Mono Channel).

[7] conventionally, hours of speech is required

3.3 Implementation Details

We implement the Unit Selection based method for generating audio and try to use a pre-implemented neural network based method to generate audio. Although Hindi audio could not be generated using the latter, but we successfully generate audio using the former method.

3.3.1 Unit Selection based Concatenative Synthesis

We perform Unit selection synthesis using a large corpus of recorded speech labeled with the text being spoken (Details of the corpus in implementation details). During such a corpus creation, each recorded utterance is segmented into some or all of the following: a) individual phones, b) diphones, c) half-phones, d) syllables, e) morphemes, f) words, g) phrases, and h) sentences.

A specially modified speech recognizer set to a "forced alignment" mode with some manual correction is typically used to divide the speech corpus into segments (utterances). It uses visual representations such as the waveform and spectrogram, to divide the speech. An index of such units in the speech database is then created based on the segmentation and acoustic parameters like the fundamental frequency (pitch), duration, position in the syllable, and neighboring phones. At run time, the desired target utterance is created by determining the best chain of candidate units from the database (unit selection). This process is typically achieved using a specially weighted decision tree (HTS System uses HMM and looks at posterior probability and prior probability to decide the best chain.)

Since the output of our work would be used to generate pronunciations of a word/phrase/short sentences, we need a natural sounding voice and hence choose to build cluster units of the recorded speech data available.

3.3.2 Neural Network based RAW Audio generation

We also use pre-implemented models from around the web to reproduce TTS systems, but as quoted at many places, such systems require huge amounts of data and exorbitant amounts of time to generate even smallest of

the samples. We use a WaveNet implementation (basveeling/wavenet)[8] to generate RAW audio for a piano music dataset and generate audio using it.

Due to various errors in the implementation when trying to use it to generate audio based on text, we could not use this implementation for any form of Text-to-Speech generation.

3.3.3 Other Experiments

We also use other pre-trained voices available on the FestVox website to generate audio for comparison with the audio generated via our voice model. We downloaded the following voices:

1. Hindi - Male Voice,
2. Hindi - Female Voice,
3. Marathi - Female, and
4. Marathi - Male.

We generate audio using these voices. A brief record of our survey of various speech synthesis systems available is provided in Table 1. We also use the default Festival diphone-based voice for Hindi provided with the system for comparison. We also survey the other potential speech synthesis frameworks and list them in the table for reference.

Technique	Explored	Voice Models Generated	Usable for Hindi TTS
Festival+FestVox (IndicTTS Data)	***Yes***	***Many***	***Yes***
Flite Voice (Hindi - Female)	Yes	1	Yes
Flite Voice (Hindi - Male)	Yes	0	Yes
Flite Voice (Marathi - Female)	Yes	0	Yes
Flite Voice (Marathi - Male)	Yes	0	Yes
Festival (diphone)	Yes	1	Yes
Wavenet (basveeling)	Yes	1	No
DeepVoice	Yes	0	No
Merlin	No	0	No
MaryTTS	No	0	No
Tacotron	No	0	No
SampleRNN	No	0	No
Char2Voice	No	0	No

Table 1: Our tryst with Speech Synthesis- An overall picture of the area explored

[8] https://github.com/basveeling/wavenet

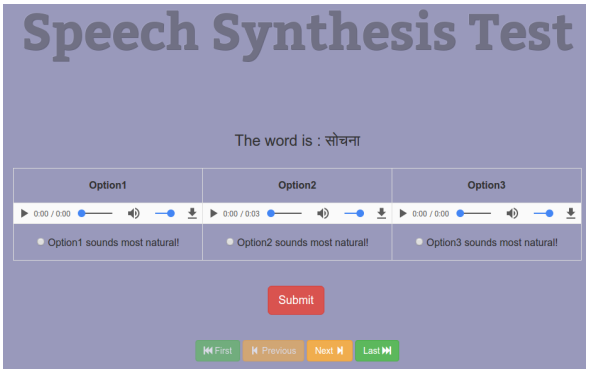

Figure 2: A Unit selection based Concatenative Speech Synthesis System

4 Results & Evaluation

We accumulate 6 usable voice models and produce word audios and randomly sample word audios from them. Among these models, the one which we successfully generated using Unit Selection based Concatenative Synthesis, sounded most natural in a brief overview.

Speech Synthesis evaluation is a subjective issue. Different speech voices are used to train various speech systems, and no agreed upon metric for the quality of such an output has been produced, yet. Quality of production technique is another factor on which speech synthesis depends, and hence the evaluation of speech synthesis systems has been compromised by differences between such factors (production techniques, recording facilities etc.)

Speech Synthesis systems require human annotators for evaluation of their output. The annotation is done based on naturalness and intelligibility of the output. A recent work proposes a novel approach that formulates objective intelligibility assessment as an utterance verification problem using hidden Markov models, thereby alleviating the need for human reference speech (Ullmann et al., 2015). Although nothing exists to assess the naturalness of a speech synthesis output.

We generate word audios for approximately 4000 words using four best voice models. For evaluation of our synthesized data, we create an experiment vaguely based on Turing Test. We randomly choose 30 Hindi Words and also get audio recorded for them with the help of our lexicographers.

We create a PHP-MySQL based web-

	#0	#1	#2	#1+#2	Most Liked
Model 1	79	55	99	154	**101**
Model 2	37	78	112	190	90
Model 3	72	86	58	144	51
Model 4	**55**	**117**	**107**	**224**	70

Table 2: Results of manual evaluation of synthesized speech clips

interface show as a screenshot in Figure 2 and crowd-source results. The interface shows a user, three different audio samples, and they were asked to choose the "Most Natural" audio from among them.

We receive a total of 442 responses for 30 word samples. Thus, we assume that 14 people had completed the test. The results of our initial evaluation based on naturalness are as follows: (i) **The mean of our voice model win percentage is over 44%**. We beat both the other voices by an acceptable margin, (ii) **Pre-recorded speech by humans was rated best somewhat less than 30%** of the times, and (iii) Grapheme based synthesized speech **scored around 26% on this scale.**

We randomly chose 535 words and generate synthesized outputs from four best models; these outputs were presented to two lexicographers for further analysis. They used the following scale to report the output (i) **unusable (#0):** This rating corresponds to audio clips which are either distorted, or too noisy for the user to comprehend, (ii) **usable (#1):** This rating corresponds to audio clips which are moderately usable and suggests that the user can comprehend the underlying words. However, audio clips with this rating can be synthesized better, (iii) **good (#2):** This rating corresponds to audio clips that are really good and convey the words. For each of the 535 words, the lexicographers were also asked to mark which of the four synthesized clips they liked the most.

The evaluation results are shown in Table 2 which clearly show that **Model 1** was marked as the most liked audio clip most often, while **Model 4** performed the best in terms of producing the most number of usable audio clips (obtained by summing clips with ratings #1 and #2).

A qualitative analysis of the synthesized clips highlighted the following issues, particularly with respect to the clips that were marked "unusable": i) Flap or tap sounds (ड़, ढ़) were pronounced incorrectly, ii) Intonation of the audio for heavy syllables was at times incorrectly rendered and for words such as 'एकदम', the pronunciation had a specific stress pattern which should have ideally been neutral, thus making it sound unnatural, iii) There were also a few examples of unnecessary lengthening of a vowel. For example, in बीमारी (*beemari*, sickness), there was unnecessary stress on 'बी' and hence it was lengthened, iv) Incorrect syllable breaks were observed in some words. For example, नापसंद (*naapasand*, non-favourite),was pronounced as नाप–संद, which is incorrect, v) It was also noted that sometimes consonant clusters were mispronounced. E.g. कुत्ता - (*kutta*) - dog, was incorrectly pronounced as कु–ता or कुत–ता.

Eventually, we employ the best voice model for generating word, gloss, and example audios. **We generated, using Unit Selection based Concatenative Synthesis, audios for 151831 words, and 40337 synset glosses/example sentence.**

5 Conclusion and Future Work

We present our work on generating voice models using the Festival speech synthesis system. We also describe our efforts to use deep learning based implementations for generating such a model. A survey of the current state-of-the-art techniques available for speech synthesis was also done. We download pre-generated voice models available for Hindi and provide a detailed qualitative and quantitative analysis by comparing them with the voice model generated by us. We evaluate our model via crowd-sourcing and select the best voice model to generate audios for all words, glosses and example sentences for Hindi WordNet. We believe our work will help students and language learners understand the Hindi language, and help them pronounce it as well.

In future, we plan to improve the voice model by analyzing the speech output and incorporate more data for training. We also plan to implement WaveNet and other such neural network based techniques for raw audio generation and training models to produce speech for a given text.

References

Jatin Bajaj, Akash Harlalka, Ankit Kumar, Ravi Mokashi Punekar, Keyur Sorathia, Om Deshmukh, and Kuldeep Yadav. 2015. Audio cues: Can sound be worth a hundred words? In *International Conference on Learning and Collaboration Technologies*, pages 14–23. Springer.

G. Bengu, Guyangu Liu, Ritesh Adval, and Frank Shih. 2002. Educational application of an online context sensitive dictionary. *The 17th International Symposium on Computer and Information Sciences*.

P Bhattacharyya. 2010. Indowordnet. lexical resources engineering conference 2010 (lrec 2010). *Malta, May*.

Alan Black. 1997. The festival speech synthesis system: System documentation (1.1. 1). *Technical Report HCRC/TR-83*.

Christiane Fellbaum. 1998. *WordNet*. Wiley Online Library.

Diptesh Kanojia, Shehzaad Dhuliawala, and Pushpak Bhattarcharyya. 2016. A picture is worth a thousand words: Using openclipart library for enriching indowordnet. In *Eighth Global WordNet Conference*. GWC 2016.

SP Kishore, Alan W Black, Rohit Kumar, and Rajeev Sangal. 2003. Experiments with unit selection speech databases for indian languages. *Carnegie Mellon University*.

Kevin Knight and Steve K Luk. 1994. Building a large-scale knowledge base for machine translation. In *AAAI*, volume 94, pages 773–778.

Richard E Mayer. 2002. Multimedia learning. *Psychology of learning and motivation*, 41:85–139.

Dipak Narayan, Debasri Chakrabarti, Prabhakar Pande, and Pushpak Bhattacharyya. 2002. An experience in building the indo wordnet-a wordnet for hindi. In *First International Conference on Global WordNet, Mysore, India*.

Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. 2016. Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*.

Hemant A Patil, Tanvina B Patel, Nirmesh J Shah, Hardik B Sailor, Raghava Krishnan, GR Kasthuri, T Nagarajan, Lilly Christina, Naresh Kumar, Veera Raghavendra, et al. 2013. A syllable-based framework for unit selection synthesis in 13 indian languages. In *Oriental COCOSDA held jointly with 2013 Conference on Asian Spoken Language Research and Evaluation (O-COCOSDA/CASLRE), 2013 International Conference*, pages 1–8. IEEE.

Anand Arokia Raj, Tanuja Sarkar, Sathish Chandra Pammi, Santhosh Yuvaraj, Mohit Bansal, Kishore Prahallad, and Alan W Black. 2007. Text processing for text-to-speech systems in indian languages. In *SSW*, pages 188–193.

Ray Richardson and Alan F Smeaton. 1995. Using wordnet in a knowledge-based approach to information retrieval.

Richard Sproat. 1996. Multilingual text analysis for text-to-speech synthesis. *Natural Language Engineering*, 2(4):369–380.

Satoshi Takano, Kimihito Tanaka, Hideyuki Mizuno, Masanobu Abe, and S Nakajima. 2001. A japanese tts system based on multiform units and a speech modification algorithm with harmonics reconstruction. *IEEE Transactions on Speech and Audio Processing*, 9(1):3–10.

Raphael Ullmann, Ramya Rasipuram, Hervé Bourlard, et al. 2015. Objective intelligibility assessment of text-to-speech systems through utterance verification. In *Proceedings of Interspeech*, number EPFL-CONF-209096.

Piek Vossen. 1998. *A multilingual database with lexical semantic networks*. Springer.

Ren-Hua Wang, Zhongke Ma, Wei Li, and Donglai Zhu. 2000. A corpus-based chinese speech synthesis with contextual dependent unit selection. In *Sixth International Conference on Spoken Language Processing*.

Heiga Zen, Takashi Nose, Junichi Yamagishi, Shinji Sako, Takashi Masuko, Alan W Black, and Keiichi Tokuda. 2007. The hmm-based speech synthesis system (hts) version 2.0. In *SSW*, pages 294–299.

Heiga Zen, Keiichi Tokuda, and Alan W Black. 2009. Statistical parametric speech synthesis. *Speech Communication*, 51(11):1039–1064.