MARY TTS unit selection and HMM-based voices for the Blizzard Challenge 2013

Marcela Charfuelan[1], *Sathish Pammi*[2], *Ingmar Steiner*[1]

[1]DFKI Language Technology Lab, Berlin and Saarbrücken, Germany
[2]ISIR, Université Pierre et Marie CURIE (UPMC), Paris, France
firstname.lastname@(dfki.de|isir.upmc.fr)

Abstract

This paper describes the implementation of a unit selection English voice and a HMM-based Hindi voice for our participation in the Blizzard Challenge 2013. The two voices have been created using the MARY TTS voice building framework. We describe how audiobook data is used to create the English voice and how a quality control measure (statistical model cost) is used to control the selection of unit candidates, in addition to target and join costs. The implementation of the Hindi voice and the new Hindi language components in the MARY TTS framework are also described. We have obtained close to average results for both systems, especially in the emotion category for the English voice, Naturalness for the Hindi voice and Word Error Rate (WER) for both systems.

Index Terms: speech synthesis, unit selection, join cost, multilingual, open source

1. Introduction

This year's Blizzard Challenge gives us the opportunity to investigate and test new ideas in expressive speech synthesis under the MARY TTS framework [1]. In unit selection synthesis, the challenge of creating voices using audiobook data gives us the opportunity to test the use of a statistical quality control measure (sCost) at the unit level in very expressive data. We were also able to support the MARY TTS framework by adding new languages, in this case, Hindi. Mainly due to the small size of the Hindi database provided, we decided to create a HMM-based voice with it, although a unit selection voice could also have been possible.

One of the challenges of building unit selection voices using audiobook material is how to avoid discontinuities at join points, especially when the data used is very expressive. Hybrid unit selection HMM-based synthesis techniques have been proposed to improve speech quality by selecting better unit candidates [2, 3]. We also used a HMM-based synthesis approach to improve the selection of unit candidates, but our approach is done off-line as a pre-processing stage, in which a statistical model cost at the unit level is calculated. The basic idea of this approach is to compare a sentence in the corpus with a sentence generated by a HMM-based voice trained with the same corpus. The comparison is done in terms of spectral parameters at the unit level. The sCost measure was developed in our previous work [4], where it was used to automatically find labelling errors, so to improve the quality of concatenation units.

The other challenge in which we participate this year is on building a Hindi voice, for doing so we have made use of the voice building tools in the MARY TTS framework.

The paper is organised as follows. First, in Section 2 we briefly describe the current status of MARY TTS in its latest release. In Section 3 and Section 4 the two created voices are described. Section 5 includes analysis of the listening test results and finally conclusions are drawn in Section 6.

2. MARY TTS 5

The MARY TTS platform is an open-source, modular architecture for building text-to-speech systems, including unit selection and statistical parametric waveform synthesis techniques [5]. The code in the latest release, MARY TTS 5.0, has been thoroughly restructured from the previous version (4.3.1); the main new features include:

- Simplified installation and voice distribution
- Agile build management and integration of MARY TTS into other projects (using Apache Maven [6])
- New MaryInterface API
- Emotion Markup Language (EmotionML) support

Details about these new features and the new modularised code can be found in the new development repository [1].

3. Building a unit selection English voice

The novelty included in the MARY TTS unit selection algorithm is the use of a statistical model Cost (sCost), in addition to target and join costs, for controlling the selection of unit candidates.

3.1. sCost calculation

sCost is a pre-computed statistical quality measure calculated in several steps:

1. *Estimation of segment labels*, based on recorded speech and phonetic transcription from text prompts. For this task we used the EHMM automatic labeller from Festvox [7].
2. *Creation of a HMM-based voice*, we use the labels obtained in the previous step, the transcriptions, and the recordings to create a HMM-based voice; the procedure for building a HMM-based voice in the MARY TTS framework is described in [8]. For building this voice we used just the provided segmented data "Mansfield Park", and in order to generate a HMM-based voice with a stable, not too expressive, narrative style, we used the same techniques as [9] to create a neutral voice from the audiobook data.
3. *Extraction of spectral parameters*, Mel generalised cepstrum (mgc) features are extracted from the recordings

using SPTK [10], and the same spectral features are generated with the HMM-based voice, in the MARY TTS framework, using the transcriptions provided. The labels (phone durations) generated for each sentence are also kept for performing the alignment in the next step.

4. *Calculation of sCost*, labels and dynamic time warping (DTW) are used to align the spectral feature vectors from the extracted and generated Mel cepstrum parameters and calculate an optimum path sCost measure. The criterion for finding the optimal path is the Mahalanobis distance between vectors, where the variance is the phone variance computed on the recorded waveforms. sCost is computed as the sum of the Mahalanobis distance over the optimal path, divided by the number of frames in the recorded segment and in the generated segment.

3.2. sCost deployment

sCost is used in the MARY TTS unit selection algorithm as follows. As usual, the unit selection algorithm includes two types of unit costs: *target cost* to define how well a unit candidate from the database matches the target unit; and *concatenation cost* to define how well two selected units can combine at joins. The cost functions can be written as follows:

$$targetCost(u_i) = \langle w, c(u_i) \rangle \quad (1)$$

$$joinCost(u_i, u_{i-1}) = \langle w, c(u_i, u_{i-1}) \rangle \quad (2)$$

where u_i is the i^{th} unit candidate; c is the cost vector; and w is the weight vector for these features. In addition to these two costs, each unit candidate is associated with a precomputed sCost. Then the overall cost for selecting units can be expressed as:

$$totalCost(u_i) = W_1^{\mathrm{T}} * \begin{pmatrix} targetCost(u_i) \\ joinCost(u_i, u_{i-1}) \\ sCost(u_i) \end{pmatrix} \quad (3)$$

Thus, at the stage of selecting units, a dynamic programming algorithm finds the best suitable candidates for the target by minimising the total cost function. Beam search is used to minimise the speed of computation. Weights for the different costs are obtained heuristically as described in Section 5.1.

Further details on the implementation of sCost is presented in [11].

4. Building a HMM-based Hindi voice

In this section the implementation of minimal NLP components for the Hindi language in the MARY TTS framework and the novelties included in the mixed excitation generation of the Hindi HMM-based voice are described.

4.1. Hindi NLP components

For Hindi language NLP components, MARY TTS takes unicode-formatted text input. A rule-based phonemiser is implemented for two purposes: (i) converting unicode sequences to the IT3 phone set [12]; (ii) schwa deletion according to the rules defined in [13].

4.1.1. Phone set

In order to process Hindi text, we initially convert Hindi unicode letters to the IT3 phone set just before applying pronunciation rules. The conversion map is as shown in Table 1. Having defined the phone set, we create an allophone file in MARY TTS format.

Vowels	Consonants
a (अ)	k (क)
aa (आ)	kh (ख)
i (इ)	g (ग)
ii (ई)	gh (घ)
u (उ)	ng∼ (ङ)
uu (ऊ)	ch (च)
rx (ऋ)	chh (छ
e (ऐ)	j (ज)
ei (ए)	jh (झ)
ai (ऐ)	nj∼ (ञ)
o (ओ)	t (त)
oo (ओ)	th (थ)
au (औ)	d (द)
a: (ः)	dh (ध)
	n (न)
	t: (ट)
	t:h (ठ)
	d: (ड)
	d:h (ढ)
	nd∼ (ण)
	p (प)
	ph (फ)
	b (ब)
	bh (भ)
	m (म)
	y (य)
	r (र)
	l (ल)
	l: (ळ)
	v (व)
	sh (श)
	shh (ष)
	s (स)
	h (ह)
	r: (ऱ)
	n: (ं)

Table 1: IT3 notation [12]

4.1.2. Schwa deletion

Schwa deletion is the main challenge in Hindi pronunciation prediction. Each consonant in written Hindi is associated with an "inherent" *schwa* (pronounced as [ə] or [ʌ]).

The problem is that schwa is sometimes pronounced and sometimes not. In this paper we have adopted the following set of rules, proposed by Choudhury [13], for schwa deletion:

1. The schwa of a syllable immediately followed by a conjugate syllable (*yuktakshara*) is always retained.
2. If *y* (य) is followed by the inherent schwa and preceded by a syllable with a high vowel such as *i, ii, rx, u* or *u*, then the schwa following *y* is always retained.
3. Any conjugate syllable or cluster of consonants that ends in (i.e. the last consonant of the cluster/syllable is) *y, r, l* or *v*, retains the schwa following the cluster.

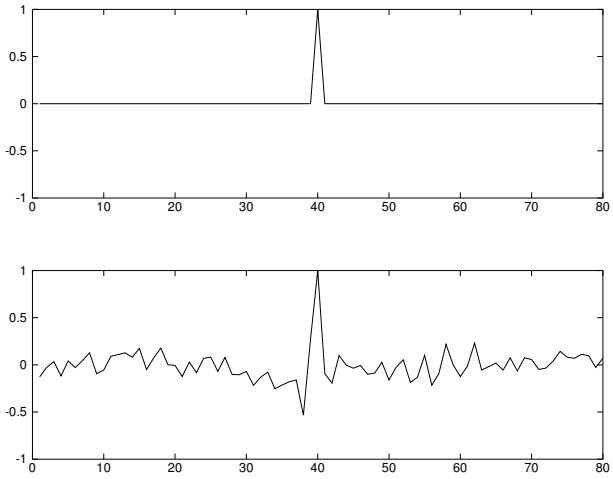

Figure 1: Top: a centred pulse in a window of size 80 samples. Bottom: a real glottal pulse obtained by inverse filtering, also centred in the window and normalised.

4. The schwa preceding a full vowel is retained to preserve lexical distinctions.
5. The schwa of the first syllable is never deleted.
6. If the last syllable of the word contains a schwa and the contexts 1 through 5 described above for the retention of the schwa do not occur, then the schwa is deleted.

Choudhury [13] claims that the performance of the above rules is 96.12%. According to him, morphological analysis before application of these rules improved the performance of the pronunciation module to 99.89%. However, the Hindi TTS system presented in this paper does not perform any morphological analysis.

With these components it is then possible to extract acoustic and context features from the audio and text files, which is used to train a unit selection or HMM-based voice. Due to the size of the audio database provided for Hindi we decided to create a HMM-based voice.

4.2. Improved mixed excitation

Once the new Hindi NLP components are created in the MARY framework, we use the usual procedure for creating HMM-based voices. In this version however we have included a preliminary version of an improved glottal model for mixed excitation. As in the normal HMM training procedure, fundamental frequency (f0) and Mel generalised cepstrum features (mgc) are extracted; we also extract voicing strengths (str) estimated by peak normalised cross-correlation of the speech filtered in five bands. During synthesis, f0, mgc, and str features are generated; as in [14], str features are used to generate two shaping filters. Shaping filters for pulse hp_j and noise hn_j are obtained from the generated str and the original bank of filters h_{ij}:

$$hp_j = \sum_{i=1}^{N} str_i * h_{ij} \quad (4)$$

$$hn_j = \sum_{i=1}^{N} (1 - str_i) * h_{ij} \quad (5)$$

where N is the number of filtered bands and j is the number or taps in each filter.

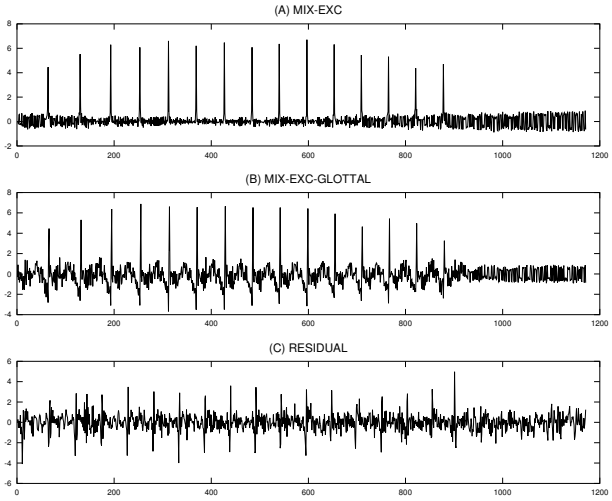

Figure 2: Section of the speech excitation (source) generated for a IH1.1 sentence: (A) mixed excitation using a simple pulse and (B) mixed excitation using a real glottal pulse obtained by inverse filtering. (C) original residual signal obtained by inverse filtering.

Final mixed excitation is obtained by adding the result of applying these two shaping filters to a pulse signal and a noise signal.

For the Hindi voice, instead of using a simple pulse, we used a (normalised) real glottal pulse obtained by inverse filtering from a speech sample (see Figure 1). The noise signal is a uniformly distributed random signal.

In Figure 2 we can observe an example of mixed excitation using a simple pulse (A), a real glottal pulse (B), and the original residual signal; as is readily apparent, the signal (B) is closer to the residual signal. From a perceptual point of view we found that this small change produces a sense of greater naturalness in the sound, reducing buzziness, when compared with the previous mixed excitation method. We have experimented with several pulses and selected the one which produced the best results for this data.

5. Blizzard listening test results

In the Blizzard listening test the MARY TTS system is identified by the letter E. The following are the main results for the two tasks in which we have participated: EH2 and IH1.1.

5.1. English EH2 task

As mentioned above, the weights for target and join cost as well as sCost, were tuned manually. The weight for sCost was kept low, because it was observed that a higher value did not contribute to improve the overall quality. Artifacts were still perceived, although the speech sounded more expressive when compared with the same speech synthesised with a similar system that did not include sCost. This might explain the good rating (very close to average) on the listening test for the emotion level in paragraphs (see Figure 3). We have also observed that sCost contributed to reduce the average consecutive length (ACL) in the system, which might explain the appearance of more artifacts and the lower mean opinion score (MOS) for naturalness (see Table 2).

For the final version, it was decided to reduce the weight of

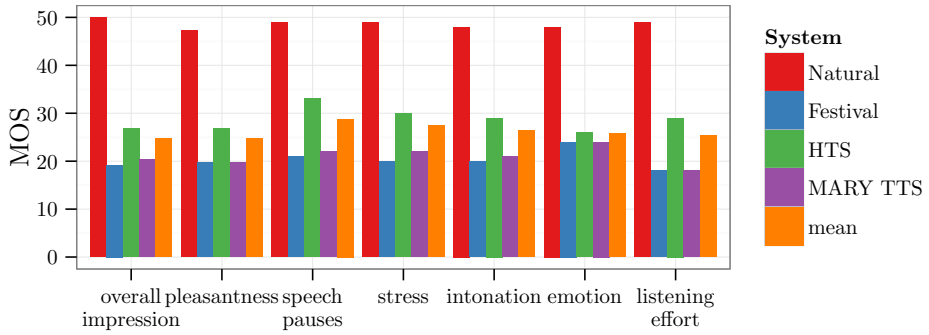

Figure 3: EH2 mean opinion score, all listeners, for paragraphs. Rank of MARY TTS system among Festival and HTS benchmark systems, mean of all systems and natural voice.

	EH2		IH1.1	
Measure	Avg.	MARY TTS Unit selection	Avg.	MARY TTS HMM-based
MOS Similarity	2.7	2.5	2.7	2.2
MOS Naturalness	2.8	2.3	3.1	2.8
WER (%)	29.8	31.0	53.0	54.0

Table 2: MOS: Mean opinion score for similarity to original speaker and naturalness for all data and all listeners; WER: Word error rates for intelligibility test, all listeners.

sCost just to the point that allowed us to avoid outliers. That is, sCost will prevent the selection of units from sentences spoken in a very different style. In further experimentation with sCost [11], (not included in the Blizzard submission due to lack of time), it was found that calculating sCost not only for the spectrum but also for fundamental frequency (f0) and voice strengths (str) gives better results in terms of reducing artifacts, increasing ACL and keeping a more stable style.

On the other dimensions for paragraphs, our system was rated below, but close to average. Figure 3 shows a comparison with natural speech, other systems (mean of all systems) and the two benchmarks Festival and HTS. As can be seen in this Figure, our system is fairly close to the Festival benchmark, but still not close enough to HTS; that other benchmark, a HMM-based system, was rated above average in all dimensions. MARY TTS ratings for MOS similarity and WER are also close to average, as shown in Table 2.

5.2. Hindi H1.1 task

The main results for our Hindi voice are presented in Table 2. We have obtained relatively better results for MOS naturalness and WER than for MOS similarity to the original speaker. This could be due to the muffled and buzzy effect of the HMM-based voice, which we are still in process of improving. Our simple improvement in mixed excitation generation, however, seems to be moving in the right direction given the ratings for naturalness.

We are using simple Hindi NLP components without any morphological analysis. Better morphological analysis would improve the rule-based pronunciation used in our system.

6. Conclusions

This paper has described two voices built using the MARY TTS framework to participate in the Blizzard Challenge 2013. The voices were created using two technologies and languages within the same framework: unit selection English and HMM-based Hindi.

For the unit selection English voice, created using audiobook data (task EH2), we have made use of the English language components and resources available in the MARY TTS framework. The novelty in this case was the use of a quality control (sCost) measure in addition to target and join costs for selection of unit candidates. sCost was used mainly to remove outlier units (units from very different styles), yet it did not help that much to improve the overall quality. The results were close to average, in particular for emotion dimension in paragraphs and WER.

For the HMM-based Hindi voice, first of all, we needed to create NLP components for this language in the MARY TTS framework. The addition of these components proved that the voice import tools included in the framework are robust enough and relatively simple to use. The results obtained for Hindi were close to average in MOS naturalness and WER.

Given the results in both systems we can consider the two synthesis technologies in MARY TTS stable enough to continue research on state-of-the-art expressive speech synthesis. In particular, in future work we will continue to improve the excitation generation in HMM-based voices, to generate or include close to natural pulses in different expressive styles or voice qualities. In unit selection we will continue exploring the use of sCost as a way to control different levels of expressivity; we will consider training more expressive HMM-based voices in addition to calculating sCost for spectrum, f0 and voicing strengths, as we have done in [11].

7. Acknowledgements

This work is supported by the EU project SSPNet (FP7/2007-2013).

8. References

[1] "MARY TTS Development Repository." [Online]. Available: https://github.com/marytts/marytts

[2] Z.-H. Ling and R.-H. Wang, "HMM-based hierarchical unit selection combining Kullback-Leibler divergence with likelihood criterion," in *IEEE International Conference on Acoustics, Speech*

and Signal Processing, Honolulu, HI, USA, 2007, pp. IV–1245–IV–1248.

[3] A. W. Black, C. L. Bennett, B. C. Blanchard, J. Kominek, B. Langner, K. Prahallad, and A. Toth, "CMU Blizzard 2007: A hybrid acoustic unit selection system from statistically predicted parameters," in *Blizzard Challenge Workshop*, Bonn, Germany, 2007.

[4] S. Pammi, M. Charfuelan, and M. Schröder, "Quality control of automatic labelling using HMM-based synthesis," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, Taipei, Taiwan, 2009, pp. 4277–4280.

[5] M. Schröder, M. Charfuelan, S. Pammi, and I. Steiner, "Open source voice creation toolkit for the MARY TTS platform," in *Interspeech*, Florence, Italy, 2011, pp. 3253–3256.

[6] T. O'Brien, M. Moser, J. Casey, B. Fox, J. van Zyl, E. Redmond, and L. Shatzer, *Maven: The Complete Reference*. Sonatype, 2008–2013. [Online]. Available: http://www.sonatype.com/resources/books/maven-the-complete-reference

[7] *Festvox – EHMM*. [Online]. Available: http://festvox.org/

[8] MARY TTS, "VoiceImportTools Tutorial," 2012. [Online]. Available: https://github.com/marytts/marytts/wiki/VoiceImportToolsTutorial

[9] M. Charfuelan and I. Steiner, "Expressive speech synthesis in MARY TTS using audiobook data and EmotionML," in *Interspeech*, Lyon, France, 2013.

[10] S. Imai, T. Kobayashi, K. Tokuda, T. Masuko, K. Koishida, S. Sako, and H. Zen, "Speech signal processing toolkit (SPTK), Version 3.3," 2009. [Online]. Available: http://sp-tk.sourceforge.net

[11] S. Pammi and M. Charfuelan, "HMM-based sCost quality control for unit selection speech synthesis," in *ISCA Speech Synthesis Workshop*, Barcelona, Spain, 2013.

[12] L. Prahallad, K. Prahallad, and M. Ganapathiraju, "A simple approach for building transliteration editors for Indian languages," *Journal of Zhejiang University Science*, vol. 6A, no. 11, pp. 1354–1361, 2005.

[13] M. Choudhury, "Rule-based grapheme to phoneme mapping for Hindi speech synthesis," in *9th Indian Science Congress of the International Speech Communication Association (ISCA), Bangalore, India*, 2003.

[14] W. C. Chu, "Mixed excitation linear prediction," in *Speech Coding Algorithms: Foundations and Evolution of Standardized Coders*. Wiley, 2003, ch. 17, pp. 454–485.