The Speect text-to-speech entry for the Blizzard Challenge 2016

Johannes A. Louw, Avashlin Moodley, Avashna Govender

Human Language Technologies Research Group
Meraka Institute, CSIR, Pretoria, South Africa
jalouw@csir.co.za, amoodley1@csir.co.za, agovender1@csir.co.za

Abstract

This paper describes the Speect text-to-speech system entry submitted to the Blizzard Challenge 2016. The focus of this entry was to build a data driven text-to-speech system that generates an expressive voice suitable for children's audiobooks. The techniques applied for the task of the challenge and the implementation details for the alignment of the audio books and text-to-speech modules are described. The results of the evaluations are given and discussed.

Index Terms: HMM-based speech synthesis, expressive speech, audio books, Blizzard Challenge 2016

1. Introduction

This paper presents our third entry into the Blizzard Challenge, where techniques were applied for the task of building a voice suitable for children's audiobooks. Voices that are required specifically for children's audiobooks differ from that of conventional audiobooks in the sense that they are read more expressively. Thus, a suitable voice for children's audiobooks would be one that can express various emotions (eg. happy and sad), speaking styles (eg. whispering, yelling) and multiple character voices that are typically used in story telling. The training data provided to the participants consisted of recorded and annotated speech of a single speaker taken from various children's stories. In total, approximately 6 hours of unsegmented speech data was provided. The data provided was not entirely aligned, some audio files contained non-speech sounds and the accompanied text files were provided in varying formats. All the audio book alignments, and utterance and label generation modules were implemented in-house and only the supplied recordings and text annotations were used. The remainder of the paper is organised as follows, Section 2 gives an overview of our text-to-speech (TTS) system, Section 3 describes our methods for building the voices for the task, Section 4 presents the Blizzard challenge evaluation results followed by a discussion and conclusion in Section 5.

2. System Overview

The architecture of our TTS system, known as *Speect*, has been reported on in previous publications [1, 2, 3] and will be briefly repeated here. The system consists of the *engine* and *plug-ins*, where the engine 1) loads all required data for a particular synthetic voice, 2) calls and controls the flow of all the synthesis modules and 3) handles system calls and memory requirements, while the plug-ins define synthesis modules, which range from natural language processing modules (NLP) such as text normalization to digital signal processing (DSP) modules, for example waveform synthesis. This architecture allows the engine to remain independent of the language of the synthetic voice as well as the method of synthesis, requiring only the implementation of new plug-ins in order to add functionality. The internal utterance structure is represented as a *Heterogeneous Relation Graph* (HRG) [4] as is the case in Festival [5] and Flite [6]. The system was written in the C language and uses SWIG (Simplified Wrapper and Interface Generator) [7] in order to create an easily accessible application programming interface (API) in scripting languages. The following sub-sections describe the relevant changes that we have made to our system for the Blizzard Challenge.

2.1. Lexicon and pronunciation prediction

Our South African English phone set, which is based on the Festival MRPA phone set as used in Festival, and described in [2], together with our grapheme-to-phoneme (G2P) rule set was used for pronunciation prediction. For the purposes of the challenge we also added 73 entries (mainly proper names found in the test set) which did not exist in our source dictionary and where our G2P failed to produce the correct pronunciation.

2.2. Part-of-speech

Our part-of-speech (POS) tagger was based on the simple *guessed* POS tagger in Festival, which contains a closed list of class words and all other words are assumed to be *content* class words. The motivation for using such a naive approach was that we wanted to test the rest of the NLP based modules using a POS tagger that we would be able to duplicate in resource scarce languages.

2.3. Classification and clustering

One of the key challenges to building an expressive text-to-speech system is the annotation of the expressiveness in the speech corpus. In most approaches, a separate voice is trained to represent each emotion and speaking style. When using speech taken from audiobooks this is a more difficult task as the speech corpus contains a mixture of emotions and speaking styles. Manual annotation is typically used as there is no standard way of annotating such data. For the task of the Blizzard challenge, such a method would be impractical as it is costly and time consuming. Therefore an unsupervised clustering technique, proposed in [8] needed to be used to model the emotion and voice styles of our voice. The aim of the clustering, is to cluster the training data into expressions that have a similar nature. Each cluster is then trained separately. Looking at the text found in audiobooks, most of the expressive speech lies within direct speech, whereas the text that lies outside the direct speech are typically read in a narrative or neutral voice. Sentences can be classified into three types of units: narration, carrier and direct speech. The presence of double quotation marks were used

as markers in order to classify the text accordingly. Text found in the same sentence as the direct speech but outside the quotation marks were seen as carriers. Sentences that contained no direct speech were classified as narrative. For example: *The girl said, "I love playing in the garden because of the beautiful flowers," and then smiled to herself. Then she plucked one from the ground.*

1. Carrier: The girl said
2. Direct Speech: "I love playing in the garden because of the beautiful flowers,"
3. Carrier: and then smiled to herself.
4. Narrative: Then she plucked one from the ground.

By segmenting the text in this way, an appropriate tone of the speech can be conveyed, in this example the tone of the direct speech will be happy. For clustering, there are three primary options that can be used for unsupervised clustering which include using: acoustic data only; text data only or both the acoustic and text data. For the Blizzard task, it was decided that the data be clustered acoustically. The overall idea was to train an average voice model using all the data together and then using Hidden Markov Model (HMM) adaption [9], we would adapt the average voice model using the data found in each cluster so that each cluster would resemble a specific style of expressiveness. In this way, models will be generated based on clusters that share the same properties. At synthesis time, the text would then be classified and the corresponding cluster together with its models would be selected to generate the corresponding speech data.

2.4. Phrasing

Our current prosodic phrasing prediction is based on a simple naive model using punctuation cues in the text in order to predict where phrase breaks should be inserted in the synthesised target utterance. Our goal was to move towards data-driven techniques due to the lack of hand annotated phrase break data in our local languages, thereby exploring options we can apply in our own work. We implemented data-driven phrasing as described in [10], which builds on an earlier grammar based phrasing method done in [11]. The method combines the conditional probabilities of POS and phrase break sequence models using Bayes theorem:

$$P(b_i|C_i, B_i) = \frac{P(b_i|C_i) \cdot P(b_i|B_i)}{P(b_i)} \quad (1)$$

where $P(b_i|C_i, B_i)$ is the probability of having a phrase break at juncture i, given a context of POS features, C_i, and the context of previous break features, B_i, both observed at i. We derived acoustic phrase breaks using a similar Hidden Markov Model Toolkit (HTK) [12] based tool (as described in Section 3.3) in order to extract the natural prosodic phrase breaks of the speaker. In our model we used a conditional random field (CRF) in order to model the POS sequence and the previous phrase break sequence, whereas [10] used a stochastic context free grammar (SCFG) and a classification and regression tree (CART) respectively.

2.5. Intonation

The Blizzard Challenge in 2016 required that the voice created be expressive. One important factor that contributes to the expressiveness of a voice is being able to successfully model the intonation of an arbitrary utterance. The Automatic Stylization and Labelling of Speech Melody (SLAM) approach proposed in [13] was modified to suit the purposes of modeling the intonation for the Speect TTS System.

2.6. SLAM Modeling

The SLAM method is a data-driven approach to apply style labels to segments of an utterance [13]. SLAM allows for labeling to occur at all levels of the utterance (Phrase, Word, Syllable or Phoneme). All frequencies are converted to the semitone scale and expressed with respect to the mean frequency of the speaker, $f_{0_{st}}$ using Equation 2.

$$f_0(st) = 12 \times \log_2 \frac{f_0(hz)}{f_{0_{mean}}(hz)} \quad (2)$$

The thresholds for the semitone f_0 labels are in Table 1. The label construction occurs as follows:

1. The frequency at the start ($f_{0_{start}}$) of the segment gets converted using Equation 2. This is then measured against the thresholds to determine the label.
2. The same process is followed for the frequency at the end ($f_{0_{end}}$) of the segment.
3. Every frequency,f_{0_i}, between the start and the end will be subjected to two difference calculations, $\Delta f_{0_{start}}$ and $\Delta f_{0_{end}}$. The highest difference, $\Delta f_{0_{max}}$ will be considered when evaluating whether f_{0_i} is a peak ($f_{0_{peak}}$).

$$\Delta f_{0_{max}} = max(\Delta f_{0_{start}}, \Delta f_{0_{end}}) \quad (3)$$

If $\Delta f_{0_{max}} > 2$ semitones then the peak is significant enough to receive a label. The position of the peak in the segment is also noted (1 = start, 2 = middle, 3 = end)

The label for the segment is a concatenation of the individual labels (start label + end label + peak label), the peak label is omitted when it is not significant. For example, a possible label could be Hl or mLh2. More details on the SLAM approach can be found in [13].

2.6.1. Simplified SLAM

The SLAM approach in [13] was modified to suit the purpose of using it for intonation modeling in a TTS system. Since our TTS voices are built from data of a single speaker, the calculated mean was changed to be over the entire corpus rather than per utterance in the absence of speaker information. The semitone scale was removed and replaced by standardized f_0 values. The threshold values for the standardized f_0 labels are in Table 1. A full label is generated using the labeling technique in the original SLAM approach. The label is then simplified to be *up*, *down* or *same* based on the beginning and ending of the original label of the segment.

Table 1: Threshold labels

Labels	SLAM thresholds	Simplified SLAM thresholds
H	f >6	f >2
h	2 <f <6	1 <f <2
m	-2 <f <2	-1 <f <1
l	-6 <f <-2	-2 <f <-1
L	f <-6	f <-2

2.7. Digital signal processing

All of our Blizzard Challenge entries up to date have been statistical parametric speech synthesis (SPSS) based systems, where the vocoder used in our 2010 entry was a simple source-filter excitation model and our 2013 entry used a mixed excitation model. In this entry we explored the use of more advanced vocoders.

2.7.1. Vocoder

Initially the *Harmonic plus Noise Model* (HNM) [14] was implemented, with appropriate modifications in order to fit into the SPSS framework [15]. This was abandoned due to the difficulties in modeling the maximum voiced frequency or split between the harmonic and noise bands [16, 17], as well as the challenges in modeling the non-minimum phase terms [15]. The WORLD vocoder [18] was also investigated, but some rudimentary experiments suggests that it is relatively susceptible to voicing decision errors made by fundamental frequency prediction (f_0) models. We decided to use the *Harmonic Model + Phase Distortion* (HMPD) [19] vocoder due the fact that no voicing or band splitting decision is required, and thereby simplifying the signal representation and modeling. The HMPD vocoder decomposes a speech signal into the following signals at constant frame rates:

- $f_0(t_i)$: a fundamental frequency
- $V_i(f)$: a vocal tract filter response
- $\mu_i(t)$: a short-term mean of the phase distortion (where the phase distortion is the relative difference between two frequency components in the frequency domain)
- $\sigma_i(f)$ a short-term standard deviation of the phase distortion

Analysis and synthesis of HMPD is similar to the adaptive Harmonic Model (aHM) [20], in that the final synthetic signal ($\hat{s}(t)$) is generated by a sum of harmonic tracks:

$$\hat{s}(t) = \sum_{h=1}^{H} \hat{a}_h(t) e^{j\hat{\phi}(t)} \cdot \chi[hf_0(t) < f_s/2](t) \qquad (4)$$

where the function $\chi[hf_0(t) < f_s/2](t)$ discards harmonic content whose frequency is equal to or higher than the Nyquist frequency. Therefore HMPD is a full band model, like aHM, but the phase distortion signals ($\mu_i(t)$ and $\sigma_i(f)$) are used to recover a *synthetic* relative phase ($\check{\phi}(t)$) with the same perceived characteristics as the original one ($\hat{\phi}(t)$) [19] as follows:

$$\hat{PD}_{i,h} = \mathcal{WN}(\mu_i(hf_0(t_i)), \sigma_i(hf_0(t_i))) \qquad (5)$$

where $\mathcal{WN}(\mu, \sigma)$ generates values from a *wrapped* normal distribution around a mean μ and with a standard deviation σ. Now ($\check{\phi}(t)$) can be found as:

$$\hat{\theta}_{i,h} = \underset{h}{\Delta}^{-1} \hat{PD}_{i,h} \qquad (6)$$

where $\underset{h}{\Delta}^{-1}$ is the cumulative sum and $\hat{\theta}_{i,1} = 0$. Then adding the minimum-phase response of the vocal tract filter

$$\check{\phi}_{i,h} = \hat{\theta}_{i,h} + \angle V_i(hf_0(t_i)) \qquad (7)$$

In practice it has been found that the short-term mean of the phase distortion ($\mu_i(t)$) as done in the HMPD analysis [19] does not add any perceived benefit and therefore it has been omitted from vocoder and μ_i set to zero (0) in Equation 5.

3. Voice Building

3.1. Data

Audio books of 50 children's stories, narrated by a British English female speaker, together with the text of 40 of the stories were provided. Part of the task was to source the text of the 10 outstanding books. The duration of all the provided audio was `06:01:53.18`. The audio and text files were provided in varying formats, all at 44.1 kHz sampling rate, 2 channels, 16 bit encoding. Some of the audio files contained non-speech sounds (animated sounds of animals, laughter, notifications of page turning, etc.). Initial sentence level alignments were provided by Toshiba, where all non-speech sounds were removed and the text was segmented on what seemed like prosodic phrase breaks based on punctuation (commas, full stops, colon, exclamation, question, etc.).

3.2. Pre-processing

All audio was down-mixed to a single channel and then down-sampled to 16 kHz at 16 bits per sample. A base-line voice was built with the standard HTS [21] recipe (version 2.2) and supplied sentence level alignments in order to test the use of the South African English phone set and pronunciation prediction.

3.2.1. Text

The text was segmented on sentence level in order to fix the issues in the supplied text segmentation (where one utterance with direct and carrier speech was split into two utterances). Some of the 10 outstanding books were sourced from electronically scanned portable document format (PDF) files and required optical character recognition (OCR) software in order to extract the text. This text was then manually verified in order to eliminate errors produced by the OCR software. The final text had 5094 utterances with an average length of 8.8 words per utterance, a minimum of 1 and a maximum of 53 words per utterance.

3.2.2. Utterance level alignments

The given audio was split into page or chapter segments, these were combined into one file for each book. Utterance level alignments were then done using the text and combined audio files for each book with the `aeneas`[1] tool, which uses a dynamic time warping algorithm to force align the audio with some reference audio. The reference audio that was used was generated with the base-line voice built with the supplied alignments. All 10 outstanding books were manually verified and corrected due to mismatches of the text and what was read in the audio.

3.2.3. Pruning silences

Some of the resulting aligned utterances had large preceding and following silences, which were pruned with the Festvox [22] tool `prune_silence`. The tool executes a pitch determination algorithm (pda) and prunes parts of the initial and final wave file where it deems there is no speech (it uses pda for voice activity detection). The pruning was set at 0.2 seconds, meaning the initial and final silences should not be longer than that. If shorter, then no pruning is done. At the end we had 5079 utterances, with an pre-pruned duration of `05:44:28.20` and a pruned duration of `04:39:14.41`.

[1] https://github.com/readbeyond/aeneas/

3.3. Phonetic alignment

We use a forced-alignment process based on HTK in order to align the audio to the phonetic transcriptions of all the utterances. Speaker-specific triphone acoustic models were trained, which were then used to align the data. This same process also provides us with the acoustic phrase breaks (Section 2.4) used to train our phrasing models.

3.4. Classification and clustering

Using the approach described in Section 2.3, each sentence was classified and labeled as either direct speech, carrier or narrative. This information was then captured in the HTS labels and used as features during the HMM training. The corresponding audio was segmented accordingly. The speech data was then clustered acoustically. This was performed by first extracting low-level acoustic features on a frame level and then mapped to unit level via functionals such as mean, standard deviation etc. Using the manually selected feature sets in [8], the following eight features were extracted: mean of f_0, voicing probability (p_v), local jitter, local shimmer and logarithmic HNR; standard deviation of F_0; mean of absolute delta of f_0 and p_v. The features were extracted using the `opensmile`[2] tool. The data was then clustered using a hierarchical k-means clustering technique. 3 clusters were formed. Due to time constraints, the adaptation of the average voice model to the individual clusters was not completed in time. The integration of the adaptation procedures with Speect took longer than expected and the time required to adapt the average voice model to multiple clusters exceeded the time we had left to complete the task.

3.5. Phrase break modeling

The phonetic alignment stage (Section 3.3) outputs silences found in the read audio. All silences larger than a threshold (80 ms) were deemed to be acoustic phrase breaks. Our final set of 5079 utterances had silences in with a mean of 0.283 seconds and a standard deviation of 0.231 seconds. These silences, which are assumed to be acoustic phrase breaks, together with POS tags of the words in the utterance were used to train a CRF model (with the `CRFsuite` [3]) that provided the grammar based probability of a phrase break ($P(b_i|C_i)$ in section 2.4). Next another CRF was trained, using these predicted phrase breaks features together with word positional and punctuation features in order to provide the final phrase break prediction. Table 2 gives the objective results of the phrase break model for the acoustic breaks and non-breaks when compared to the audio on a test set (10% of data).

Table 2: Objective results for phrase break model

	Precision	**Recall**	**F1 score**
Non-break	98.04%	98.6%	98.32%
Phrase break	86.33%	81.82%	84.01%

3.6. Simplified SLAM prosody modeling

Every audio file in the corpus was processed to extract f_0 values using the MELodic MOdelisation (MOMEL) algorithm [23]. The mean and standard deviation of the corpus were calculated from these values. Textgrids were generated with alignment in place at all segment levels. Each textgrid had already been processed by the part-of-speech tagger and the phrase break model. Each segment of the textgrid was given a simplified SLAM label. The textgrids were then processed on a phrase, word and syllable level to extract contextual features for the part-of-speech tags, phrase breaks and SLAM tags. Positional features were also extracted. Three CRF models were trained (with the `CRFsuite`): a phrase level, a word level and a syllable level model. Both the phrase and word model results were used as additional features in the syllable model. This formed a cascading model which was then used for predictions. At synthesis time, a SLAM label will be assigned to all levels of the target utterance. Table 3 gives the objective results of the syllable level SLAM model for the labels up,$down$ and $same$ on a test set that consists of 10% of the data

Table 3: Objective results for the simplified SLAM syllable model

	precision	**recall**	**F1**
up	77.80%	56.23%	65.28%
down	73.89%	56.27%	63.89%
same	86.04%	94.05%	89.87%

3.7. Training

Training of HMM models was done via custom scripts based on the standard demonstration script available as part of HTS (version 2.2). HMPD features were extracted using the standard COVAREP [24] scripts at a constant 5 ms frame rate. The harmonic amplitude envelope was compressed with 39 MCEP coefficients, while the standard deviation of the phase distortion was modeled with 12 MCEP coefficients. Global variance was included. The standard linguistic questions were used for model tying, including: simplified SLAM tags on syllables (previous, current and next), and the token type (direct, narrative or carrier) feature of the word. We did not have syllable stress and therefore excluded it from the questions.

4. Results

The test sentences were a mixture of news type utterances, semantically unpredictable sentences (SUS), and sections from audiobooks, ranging from partial sentences, to paragraphs, chapters and complete books. These test sentences were synthesised and evaluated in a large online perceptual evaluation experiment. Listeners were grouped into paid participants, volunteers and speech experts. The results presented here are from all listeners due to space constraints. In the challenge the submitted entries were evaluated in the following categories:

- similarity to original speaker
- mean opinion score (naturalness),
- word error rate (intelligibility) on SUS, and
- naturalness of paragraph and longer based synthesis, which included: *emotion*, *intonation*, *listening effort*, *overall impression*, *pleasantness*, *speech pauses* and *stress*.

Our designated system identification letter for the results is "I". System "A" is natural speech, "B" is the Festival benchmark system (a standard unit-selection voice built using the same method as used in the CSTR entry to Blizzard 2007), system "C" is the HTS benchmark, system "D" is a deep neural network (DNN) benchmark and systems "E" to "Q" are participants.

[2] http://opensmile.sourceforge.net/

[3] http://www.chokkan.org/software/crfsuite

4.1. Similarity to original speaker

Figure 1 gives the perceptual evaluation result for all the listeners for the ”similarity to original speaker” evaluation. From the pairwise Wilcoxon signed rank tests we can see that our system is not significantly different (1% confidence level) from systems C, G, K and N.

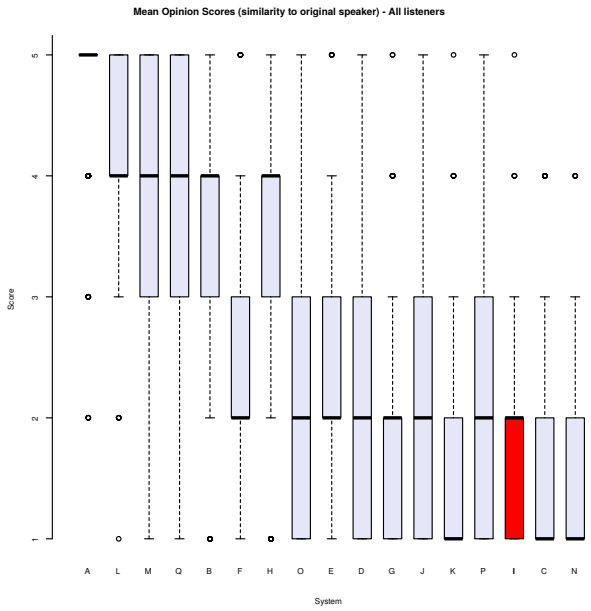

Figure 1: Similarity to original speaker, all listeners

4.2. Mean opinion score

Figure 2 gives the perceptual evaluation result for all the listeners for the mean opinion score (MOS) ”naturalness” evaluation. Our system is not significantly different from systems C, J, K and O.

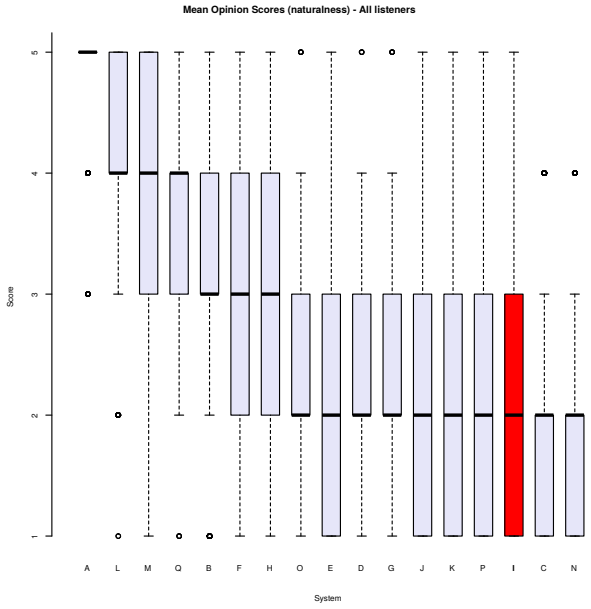

Figure 2: Mean opinion score (naturalness), all listeners

Figure 3 shows the perceptual evaluation result for all the listeners for the mean opinion score (MOS) ”overall impression” of the audiobook paragraphs evaluation. Our system is not significantly different from systems C, E, J and P.

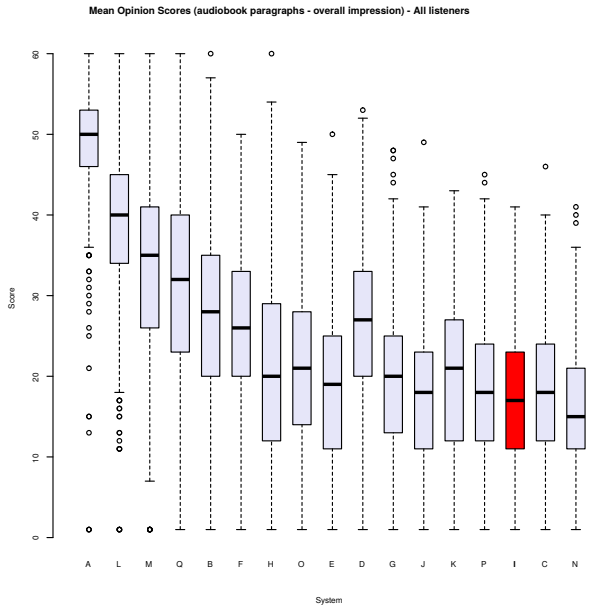

Figure 3: Mean opinion score, audiobook paragraphs (overall impression), all listeners

4.3. Word error rate

Figure 4 shows the perceptual evaluation result for all the listeners for the word error rate of the SUS evaluation. Our system is not significantly different from systems B, C, E, G, J, K, M, O and Q.

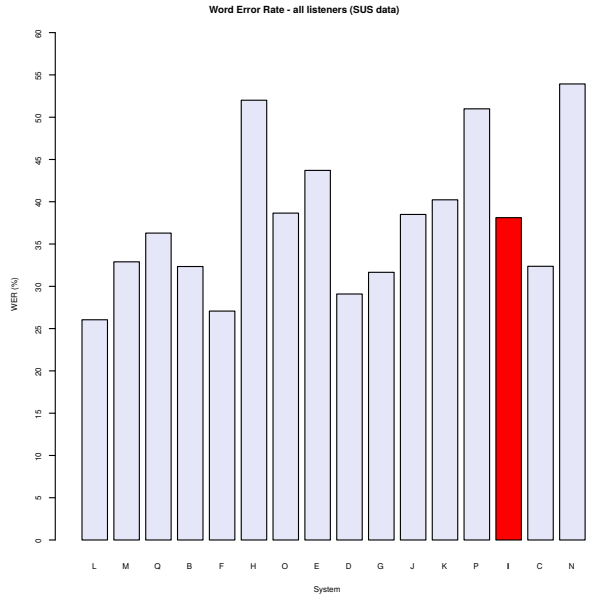

Figure 4: Word error rate (intelligibility) , all listeners

5. Discussion and conclusion

Our system suffers from a lack of naturalness and similarity to the original speaker in comparison to other systems, but intelligibility is comparable to some of the better systems. This trend was also found in our previous Blizzard Challenge entries. Although the HMPD vocoder improves the overall synthetic quality, it has inherent limitations in modeling high pitched voices [19], which is prevalent in all sinusoidal based vocoders. We also think that we might gain in naturalness by spending future efforts on DNN-based acoustic modeling. Our phrasing models worked well and are well suited to data-driven techniques. We are planning on exploring our simplified SLAM version in more detail in the future in order to fully understand it's effects on synthesised speech, and especially in tonal languages such as found in South Africa.

6. References

[1] J. A. Louw, "Speect: a multilingual text-to-speech system," in *Proceedings of the Nineteenth Annual Symposium of the Pattern Recognition Association of South Africa*, Cape Town, South Africa, November 2008, pp. 165–168.

[2] J. A. Louw, D. R. Van Niekerk, and G. I. Schlünz, "Introducing the Speect speech synthesis platform," in *Blizzard Challenge Workshop 2010*, Kyoto, Japan, September 2010.

[3] J. A. Louw, G. I. Schlünz, W. Van der Walt, F. De Wet, and L. Pretorius, "The Speect text-to-speech system entry for the Blizzard Challenge 2013," in *Blizzard Challenge Workshop 2013*, Barcelona, Spain, September 2013.

[4] P. Taylor, A. W. Black, and R. Caley, "Heterogeneous relation graphs as a mechanism for representing linguistic information," *Speech Communication*, vol. 33, no. 1, pp. 153–174, 2001.

[5] ——, "The architecture of the Festival speech synthesis system," in *3rd ESCA Workshop on Speech Synthesis*. Jenolan Caves, Australia: International Speech Communication Association, 1998, pp. 147–151.

[6] A. W. Black and K. A. Lenzo, "Flite: a small fast runtime synthesis engine," in *4th ISCA Tutorial and Research Workshop (ITRW) on Speech Synthesis*, Perthshire, Scotland, 2001, pp. 157–162.

[7] D. M. Beazley, "SWIG: An easy to use tool for integrating scripting languages with C and C++." in *4th Tcl/Tk Workshop*, Monterey, California, 1996.

[8] F. Eyben, S. Buchholz, and N. Braunschweiler, "Unsupervised clustering of emotion and voice styles for expressive tts," in *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2012, pp. 4009–4012.

[9] J. Yamagishi, T. Kobayashi, M. Tachibana, K. Ogata, and Y. Nakano, "Model adaptation approach to speech synthesis with diverse voices and styles," in *Acoustics, Speech and Signal Processing, 2007. ICASSP'07. IEEE International Conference on*, vol. 4. IEEE, 2007, pp. IV–1233.

[10] A. Parlikar and A. W. Black, "Data-driven phrasing for speech synthesis in low-resource languages," in *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Kyoto, Japan: IEEE, 2012, pp. 4013–4016.

[11] P. Taylor and A. W. Black, "Assigning phrase breaks from part-of-speech sequences," *Computer Speech and Language*, vol. 12, pp. 99–117, 1998.

[12] S. Young, G. Evermann, M. Gales, T. Hain, D. Kershaw, X. Liu, G. Moore, J. Odell, D. Ollason, D. Povey *et al.*, "The HTK book," *Cambridge University Engineering Department*, vol. 3, p. 175, 2002.

[13] N. Obin, J. Beliao, C. Veaux, and A. Lacheret, "SLAM: Automatic Stylization and Labelling of Speech Melody," in *Speech Prosody*, Ireland, May 2014, pp. 246–250. [Online]. Available: http://hal.upmc.fr/hal-00968950

[14] Y. Stylianou, "Applying the harmonic plus noise model in concatenative speech synthesis," *IEEE Transactions on Speech and Audio Processing*, vol. 9, no. 1, pp. 21–29, 2001.

[15] D. Erro, I. Sainz, E. Navas, and I. Hernaez, "Harmonics plus noise model based vocoder for statistical parametric speech synthesis," *IEEE Journal of Selected Topics in Signal Processing*, vol. 8, no. 2, pp. 184–194, 2014.

[16] T. Drugman and Y. Stylianou, "Maximum voiced frequency estimation: Exploiting amplitude and phase spectra," *IEEE Signal Processing Letters*, vol. 21, no. 10, pp. 1230–1234, 2014.

[17] J. A. Louw, "A straightforward method for calculating the voicing cut-off frequency for streaming HNM TTS," in *Pattern Recognition Association of South Africa and Robotics and Mechatronics International Conference (PRASA-RobMech), 2015*. Port Elizabeth, South Africa: IEEE, 2015, pp. 252–257.

[18] M. Morise, F. Yokomori, and K. Ozawa, "World: a vocoder-based high-quality speech synthesis system for real-time applications," *IEICE TRANSACTIONS on Information and Systems*, vol. E99-D, no. 7, pp. 1877–1884, 2016.

[19] G. Degottex and D. Erro, "A uniform phase representation for the harmonic model in speech synthesis applications," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 2014, no. 1, p. 1, 2014.

[20] G. Degottex and Y. Stylianou, "Analysis and synthesis of speech using an adaptive full-band harmonic model," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 10, pp. 2085–2095, 2013.

[21] H. Zen, T. Nose, J. Yamagishi, S. Sako, T. Masuko, A. W. Black, and K. Tokuda, "The HMM-based speech synthesis system (HTS) version 2.0." in *The 6th International Workshop on Speech Synthesis*. Bonn, Germany: Citeseer, 2007, pp. 294–299.

[22] G. K. Anumanchipalli, K. Prahallad, and A. W. Black, "Festvox: Tools for creation and analyses of large speech corpora," in *Workshop on Very Large Scale Phonetics Research, UPenn, Philadelphia*, 2011.

[23] D. Hirst, A. Di Cristo, and R. Espesser, "Levels of representation and levels of analysis for the description of intonation systems," in *Prosody: Theory and experiment*. Springer, 2000, pp. 51–87.

[24] G. Degottex, J. Kane, T. Drugman, T. Raitio, and S. Scherer, "Covarep- a collaborative voice analysis repository for speech technologies," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2014, pp. 960–964.