The IBM Submission to the 2008 Text-to-Speech Blizzard Challenge

Raul Fernandez [1], *Zvi Kons* [2], *Slava Shechtman* [2], *Zhi Wei Shuang* [3]
Ron Hoory [2], *Bhuvana Ramabhadran* [1], *Yong Qin* [3]

[1] IBM T.J. Watson Research Center, Yorktown Heights, New York, U.S.
[2] IBM Haifa Research Lab, Haifa, Israel
[3] IBM China Research Lab, Beijing, China
[1]{fernanra,bhuvana}@us.ibm.com
[2]{zvi,slava,hoory}@il.ibm.com
[3]{shuangzw,qinyong}@cn.ibm.com

Abstract

The 2008 Blizzard speech synthesis challenge provided participants with an opportunity to evaluate their systems in UK English and Mandarin. This paper describes the work behind three IBM systems submitted to the challenge for these two languages. The systems presented are concatenative unit-selection text-to-speech synthesis systems consisting of a core algorithmic base, as well as some algorithmic variants introduced not just to address the language-specific component of the synthesis engines (i.e., text-processing front-end) but also to better serve the different properties of different language types (i.e., tonal nature of Mandarin). The resulting systems were evaluated with several tasks designed to address issues like overall naturalness, intelligibility and the preservation of speaker identity. All the IBM systems submitted achieved very good performance in the two languages across the different tasks reported in this paper.

Index Terms: speech synthesis, unit selection.

1. Introduction

The Blizzard Challenge was conceived in order to better understand and compare research techniques in building corpus-based speech synthesizers. Now in its fourth year, the challenge aims to circumvent the limitation of establishing comparisons across techniques presented in the literature given that these are commonly developed on different languages, different datasets, and evaluated using variable procedures. To this effect, the organizing committee of the challenge releases one or more fixed databases to the participants with a series of task definitions, and collects the output of the systems to evaluate under a common framework.

For the year 2008 installment, the challenge provided a database of approximately 15 hours of UK English recorded by a male speaker, and a database of approximately 6 hours of Mandarin spoken by a female speaker, both recorded at 16kHz. Annotations for the English corpus consisted of *utterance*-files for each recording containing the output of the Festival text processing module, as well as a breakdown by domain of the following subcorpora making up the recording script: (1) Arctic (the phonetically-balanced Arctic dataset [1]); (2) Carroll (dialog-rich extracts from stories by Lewis Carroll); (3) Unilex (isolated words, selected for phrase-boundary coverage); (4) Emphasis (carrier sentences containing emphasized words); (5) Addresses; (6) Spelling; (7) News (sentences from the British press, such as the *Herald*). Additionally, the Unisyn Lexicon containing regional variations of English lexical items was made available by the organizers. Annotations for the Mandarin corpus contained the scripts and automatically generated Pinyin transcripts of the recordings.

The tasks consisted of building one voice dataset for Mandarin using the entire corpus, and two voice datasets for English –one using the full set of recordings and one using only the Arctic subset, which we will refer to as the English Voice A and Voice B respectively. In this paper we describe the IBM submissions to each of these categories.

2. System Description

2.1. Building the Concatenative Database

2.1.1. English

Two datasets were produced for English, as required by the specifications of the challenge, using the full and Arctic sets of the corpus. In the description that follows, any type of processing that operated on a single utterance at a time (e.g., any type of signal processing; front-end text processing) was performed only once for the entire corpus and reused when building the smaller set. However, any type of batch processing where the amount of data used influenced the outcome (e.g., alignments; building prosody trees) was performed separately for each set. Following this, we obviate making the distinction when unnecessary: by dataset we mean a relevant set.

Signal Processing: To conform to the standard IBM voice-building process, the speech waveforms were first upsampled to 22kHz. The rest of the voice-building and run-time synthesis described here is performed at this rate; final submissions were downsampled to 16kHz to conform to the specifications of the challenge.

The English utterances often contained fairly long leading and trailing silences, something which was observed to degrade the quality of alignment against the silence models. To reduce this effect, the waveforms were trimmed by monitoring short-time energy values, and eliminating leading and trailing portions of the speech that continuously fell under a threshold value. The resulting speech waveforms were then high-pass filtered at 75 Hz to reduce any potential background and electrical noise, and the instances of glottal closure were automatically detected and saved for the waveform generation step at run-time synthesis [2].

Alignments: Of the information provided in the Festival utterance files for the English dataset, only the script item was used to align the acoustic waveforms against; all other information (like phone-level alignment) was discarded. The IBM TTS system for English uses a third-of-phone sized unit (i.e., the portion of speech aligned with each of a usually 3-state left-to-right HMM model), and therefore requires subphonetic level alignment. Prior to aligning the waveforms, the recording script was run through a rules-based front-end to normalize the text, and predict pronunciations and prosodic structure from it. The final alignment dictionary was obtained by merging pronunciations from three different sources: (i) pronunciations predicted by the front-end, (ii) pronunciations contained in an in-house dictionary for British English containing standard Received Pronunciation (RP) baseforms, and (iii) the RP baseforms extracted from the Unisyn lexicon.

The speech was encoded as Mel Cepstral coefficients plus delta and delta-delta values, and run through several iterations of alignments, starting with an alignment against a reference pre-trained acoustic model, and building speaker- and context-dependent models from there on (see [2] for more details).

Unit Clustering: Each of the HMM-state-sized portions of speech contained in the alignment determines a synthesis token available to the synthesizer at run time. All tokens belonging to each unit type were collected and clustered using a decision tree that asks questions about the neighboring phonetic context during training, and can then be used at run-time to map a given phonetic context to a list of suitable candidates that can be searched over. The splits in the tree were induced by asking questions about a 5-phone phonetic context (middle phone, plus 2 preceding and following phones), and growth was monitored by examining log likelihood on spectral frames associated with a particular context. The final unit-clustering trees for Voices A and B contained about 37K and 9K leaves, respectively.

Prosody Models: The goal of prosody models is to predict energy, duration and pitch targets for each unit type at run time. These models were implemented also as decision trees that operate on a set of features, which generally depend on phonetic context and/or can be derived from the output of the front-end text analysis.

Energy-prediction trees were built for each phone in the inventory by pairing up the observed RMS value of each phone instance with its 5-phone phonetic context. Using this context as predictor, trees were grown to yield about 22K leaves for the Voice A and 3K leaves for Voice B.

A single decision tree was built to predict one duration value per phone unit based on the following 24-dimensional feature vector summarizing phrase-, word-, syllable- and phone-level information available from the front-end analysis:

1 Phrase type (e.g., question, statement, etc.)

2 Sentence-level prominence of word

3 Number of words to start of phrase

4 Number of words to end of phrase

5 Part of speech of word

6 Number of syllables to the end of word

7 Order of current syllable within word

8 Total number of syllables in word

9 Lexical stress of syllable

10 Identity of current phone

11 Voicing status of current phone

12 Broadclass (e.g., vowel, fricative, etc.) of current phone

13-24 Features (10-12) for the 2 preceding and 2 following phones

The duration-prediction decision trees for Voice A and B contained approximately 900 and 700 leaves respectively.

Finally, decision trees were trained to predict three pitch target values for the sonorant region of every syllable [3]. The following 23 observations (summarizing phone-, syllable-, phrase-, sentence- and utterance-level information) are gathered for each syllable, plus its two preceding and following syllables, to produce a final 115-dimensional vector used as input to the tree:

1. Number of sentences to start of this utterance
2. Number of sentences to end of this utterance
3. Number of phrases to start of this utterance
4. Number of phrases to end of this utterance
5. Sentence type
6. Number of phrases to start of this sentence
7. Number of phrases to end of this sentence
8. Number of words to start of this sentence
9. Number of words to end of this sentence
10. Phrase type
11. Phrase-level prominence of this word
12. Number of words to start of this phrase
13. Number of words to end of this phrase
14. Number of syllables to start of this phrase
15. Number of syllables to end of this phrase
16. Part of speech of this word
17. Total number of syllables in word
18. Lexical stress of syllable
19. Main vowel in syllable
20. Left sonorant in syllable
21. Right sonorant in syllable
22. Phone preceding left sonorant
23. Phone following right sonorant

All the features are usually derived by the front-end module to mimic the standard run-time scenario when the input available to the synthesizer is just the text. The only exception was the phrase boundary location, where we combined the front-end text analysis with pause detection to better capture beginning-of-phrase and end-of-phrase pitch patterns. The prediction of feature 11, a phrase-level degree of prominence, was based on properties of the text alone as well, and also exploited word-level emphasis information from the component of the Blizzard dataset consisting of sentences carrying emphasized words. Although this was not enabled at the run time, it is still beneficial when building the prosody trees to enable possible clustering of the syllables with similar emphasis level. To allow for this, the output of the front-end module was post-processed for all sentences from the *Emphasis* subset of the corpus, and the prominence value predicted by the front-end was overwritten and set to the maximum emphasis value allowed. This was done only for the words indicated as *emphasized* in the script (i.e., in all upper case).

Versions of the Full Voice: One notable characteristic of the English dataset is the wide prosodic variability, particularly in pitch range, observed for the set of sentences in the Carroll subset. The speaker produced these in a style that is markedly different from the rest of the corpus, something which allows for potential richness of expressivity but may at the same time pose a challenge in a concatenative system that needs to smooth out stylistic and prosodic jumps to achieve a desirable level of naturalness. Since it was not clear at this stage of the process whether such a corpus would enhance the quality of the synthesis, different pitch trees were built for the "full voice" condition : using the entire corpus and leaving out the subset of Carroll sentences. This tradeoff was investigated in a pre-evaluation step prior to synthesizing the final submissions and will be described in more detail in Section 3.

2.1.2. *Mandarin*

The voice-building process for Mandarin followed a very similar outline to the one described for English, and, therefore, we will focus only on those components that are different. As for English, the voice-building was done on upsampled 22kHz speech waveforms and the final evaluation entries downsampled to 16kHz prior to submission.

Of the information provided in the Festival utterance files for the Mandarin dataset, only the script item was used to automatically align the acoustic waveforms. One challenge for Mandarin is to produce grapheme-to-phoneme conversion to pick out one correct pronunciation from several candidates according to contextual information. This was carried out using a Transformation Based Learning (TBL) algorithm proposed in [4], which seeks to improve the performance of polyphones that originally have low accuracy. As for English, the initial alignment information is obtained at a third-of-phone-sized unit. However, because the IBM Mandarin TTS system uses syllable as its basic unit, we reshaped the sub-phonetic alignment into syllable-level alignments by means of phone-to-syllable conversion rules. Since we found some incorrect Pinyin in the released dataset, we manually inspected the alignments at this stage and made any necessary corrections before proceeding with the rest of the build.

Prosody Model Building:

To model prosodic phenomena, we assign prosodic structure to each utterance of the training script, and based on that, build models for predicting prosody targets and transitions. Our system takes into account three levels of prosodic structure constituents: prosodic word, prosodic phrase, and intonational phrase. Prosodic and intonational phrases are manually labeled during the build process whereas prosodic words are automatically predicted by a decision-tree model [5]. We then build a target decision tree and a transition decision tree both for pitch and for duration. To build the pitch target model, pitch target features are represented as a vector of four representative points in the syllable's log pitch contour $(p_{k,1}, p_{k,2}, p_{k,3}, p_{k,4})$ for the k^{th} syllable. For the duration target model, the observation d_k is the syllable's duration. In the case of the transition models, pitch features are represented as $(p_{k,1} - p_{k-1,4}, p_{k,2} - p_{k-1,3})$ and the duration transition observations as $(d_k - d_{k-1})$

The contextual predictor vectors for these trees is composed of three parts:

- Tone context information: Tone of the current syllable, previous two syllables, and next two syllables.
- Phonetic context information: Category of the preceding syllables final and the following syllables initial phones.
- Position context information: Position of the syllable within the prosodic word, prosodic phrase, and intonation phrase.

After building the decision trees, we train Gaussian Mixture Models (GMMs) to model the probabilistic distribution of the prosodic features for each leaf in a given tree. The GMM will be used to calculate target cost and transition cost at synthesis time.

2.2. Run-Time Synthesis

For synthesizing the samples we used the IBM server concatenative TTS system [2]. This system exists in two varieties, a general version used to synthesize most languages (e.g. English, other European languages, and Japanese) and a version containing algorithmic variations motivated by the tonal nature of Mandarin. Both systems share a core algorithmic base, but also contain some notable differences, among these:

- unit type: the general version uses a third-of-phone-sized unit as its basic unit type, while the Chinese works with whole syllables.
- cost function: there are some differences in the cost function which is used for the segments selection, as explained below.
- signal processing: the Chinese version does not perform any signal manipulation on the selected segments.

2.2.1. English

The English TTS starts with the rule based front-end processing. This stage converts the text into its phonetic representation and also produces additional information such as phrase boundaries, parts-of-speech and syllable stress. Each phone is then divided into 3 sub-phonetic units, and each unit is assigned to a leaf in the decision tree (sec. 2.1).

For each unit we predict the target pitch, duration and energy using the decision trees and the statistical models that we created when the voice was built. For instance, the predicted pitch values for each subphoneme unit are calculated by linear interpolation of three-per-syllable mean log-pitch values, stored in each pitch tree leaf.

We then proceed to search for the best unit sequence where the candidates are all the speech segments that belong to the required context leaf. We also augment the candidate list with candidates making up the same orthograph in the training corpus, even if they are from different leaves in the unit clustering tree, in order to allow retrieving alternate pronunciations to those prescribed by the front end (e.g., *aI-D-@* vs *i:-D-@* for word *either*).

The search for the optimal unit sequence is based on a cost function. This function is composed of two parts, the first one is the target cost which measures the deviation of the unit's pitch, duration and energy from the targets. The second part is the transition cost that measures how well two segments concatenate to each other. This includes a pitch transition cost that measures the difference between the pitch of the segments at their edges, and a spectral distance which is calculated on two short windows from each segment just before and after the edge. In addition, a penalty can be added when concatenating any two segments that were not contiguous in the original recording.

After the optimal unit sequence has been selected, we recompute the final prosody in order to minimize signal modification (such as pitch or duration modifications). The new pitch curve is a smoothed curve of the original pitch from the concatenated units. The durations of the units are left unchanged. For long contiguous speech segments we just smooth the pitch curve around the concatenation points while the middle waveforms are left unchanged [2]. The pitch modifications are done by PSOLA using the pitch marks. Before the final concatenation of non-contiguous segments, we align them by calculation of their relative offset, which maximizes the absolute value cross-correlation at the overlapping region. This helps us fix any pitch marks misalignment and waveform polarity flips.

The current voice corpus had a number of peculiarities that led us to adjust our run-time algorithms to improve the resultant TTS quality. First, both volume and voice quality (e.g., creaky voice, aspirated voice, shouting, etc.) fluctuations between different parts of the database were relatively high (e.g. Carrol corpus vs. Herald corpus). Second, the low pitch and hoarseness in this voice made it very sensitive to pitch modifications. Third, some phonemes (e.g., liquids) appeared to be extremely sensitive to concatenations, especially when selecting short non-contiguous segments. A couple of adjustments have been introduced to overcome those peculiarities:

- **Gain adjustment**: The target cost function was increased for segments that were much louder than the target. When concatenating the segments each phone was normalized to its median energy value while confining the maximal and minimal gain modification. A smooth gain curve was applied to the whole utterance.
- **Continuity improvement**: An additional phoneme-dependent penalty was added. The penalty was applied when a sequence of short non-contiguous segments were selected. In addition, a phone-dependent transition cost was added to weigh continuity more highly for classes like vowels and especially liquids (and less so for fricatives and other consonants).
- **Waveform modification minimization**: The segment's pitch was modified only if the pitch gap at the concatenation point was too large ($>7\%$); otherwise, the segments were just overlapped and added over one pitch period each on both sides of the boundary.

2.2.2. Mandarin

Text Processing: At run time, phone sequences are determined by the TBL algorithm previously mentioned. Additionally, prosodic-word, prosodic-phrase, and intonational-phrase constituency are all predicted by decision trees [5]. The prediction process is hierarchical: the lower-level prosodic boundaries are detected first, and then the higher level prosodic boundaries are detected based on the lower-level prosodic units. The prosodic structure decision tree is trained on a previously manually labeled corpus of 20000 sentences [6].

Unit Selection: In this procedure, appropriate candidates of the synthesis unit sequence are selected from the speech corpus. Dynamic programming is used to search the corpus after the target cost and transition cost are defined. We generate the context feature vector based on the Pinyin sequences and estimated prosodic structures. To obtain target and transition costs for both pitch and duration, we traverse the corresponding decision tree to a specific leaf according to the context feature vector. The GMM associated with that leaf is retrieved and used to calculate the probability of each possible candidate P; $\log(1/P)$ is then used as the corresponding cost in the search function. In addition to this prosodic cost, we also add a phonetic-context target cost to the overall target cost function based on the preceding syllables final and the following syllables initial phones, which we call *phone similarity cost*. This cost can be computed by means of a table, also trained on IBMs 20000 sentences corpus, containing the average spectral distance between each phone pair. Suppose the previous phone and the next phone of a unit S^t are P^t_{prev} and P^t_{next} and those of a candidate syllable are P^c_{prev} and P^c_{next} respectively. Then the phone similarity cost is given by $d(P^t_{prev}, P^c_{prev}) + d(P^t_{next}, P^c_{next})$, where $d(\bullet, \bullet)$ is th distance between a phone pair stored in the similarity table.

3. Pre-Evaluation for English

Since the evaluation allowed only one system per voice category, it was necessary to arrive at one final system configuration to produce the synthesis samples submitted to the formal evaluation. In the case of the English Voice A, in particular, this meant exploring whether the notable stylistic and prosodic variability that we have already discussed enhanced the overall quality. To carry out all prior testing, we first designed a development set of sentences that reflected the known domains of synthesis for the final test sentences. We chose to focus on samples of conversations, news, and stories instead of optimizing for the semantically-unpredictable sentences (SUS) task that was also part of the test. We gathered some initial informal feedback from two competing sets of utterances that had been synthesized with and without the Carroll subset of sentences (i.e., all the Carroll utterances were left in (or out of) both the database inventory and prosody models), without noticing any emerging consensus: the Carroll set showed some expressivity that was preferred by some listeners, whereas other listeners preferred the set without. To investigate this issue more formally, we decided to carry out an in-house evaluation test to assess whether there was any significant preference for either set, by looking at the following questions:

- do listeners show a significant global preference for one system?
- do listeners show a significant preference for one system as a predictable function of content?

The second of these items addresses the case where, even if there's no preference for one system most of the time, there may be a preference on individual utterances. Synthesizing from such a system, however, would entail switching in two different systems for different sentences, which would require automatic system selection. To explore this, we trained two different trigram language models on (i) the corpus of Carroll sentences only, and (ii) the corpus of news sentences that make up a large majority of the training set, and used these language models to evaluate the perplexity of an incoming sentence and assign a tag to it reflecting whether or not it was more similar to the Carroll set. We then tried to answer the second of the questions above by looking at whether the tag produced by the language model tended to agree significantly, for any given sentence, with the system preferred by the majority of the users.

We designed an A-X-B preference test using 30 sentences from each of the two systems (10 from each of the news, conversational and novel domains), and collected responses from 8 listeners who listened to the pairs randomized across system, and across pair sequences. The language models were trained and validated (on 90% and 10% respectively) of the Carroll and news sentences that appeared in the recording script. Performance on the held-out set was around 91%. The text of each sentence in the listening pair was then tagged by the language model that gave it lower perplexity. The analysis of the responses from this test did not, however, yield any significant differences. No single voice was significantly preferred most of the time. Additionally, no significant correlation was found, at a sentence level, between the voice most often favored for a particular sentence and the corpus tag given to it by the language model. The main observation here is that the additional set of Carroll sentences does not impact the quality negatively, especially after the algorithmic developments introduced above to compensate for some of the degradation we had originally observed. Based on this, the full voice was selected to synthesize the final evaluation sentences for the Voice A submission.

Several other smaller A-X-B preference tests where held during the building and tuning stages. Those tests compared 10-15 sentences from 2 or 3 different sets and were usually taken by 5-6 listeners. We used those tests to verify that the algorithmic changes introduced to the engine, and motivated by this challenge, were indeed improving its quality.

4. Formal Evaluation and Results

The formal evaluation consisted of a 5-part listening test similar to the one conducted for the Blizzard 2007 Challenge and detailed in [7]: a speaker-similarity section to judge how the synthetic speech matches the natural target (Section 1); a system pairwise comparison to judge how like or unlike 2 given systems are in naturalness (Section 2); a mean-opinion-score test from the news (Section 3) and novel (Section 4) domains; and finally a semantically-unpredictable- sentence (SUS) task (Section 5) to judge intelligibility. The results of Section 2 were not made available to the participants prior to the workshop. Three benchmarks were included in the listening tests: System A, natural speech; System B, the Festival unit-selection system built using the same method as the CSTR entry to the Blizzard 2007 Challenge; and System C, an HMM speaker-dependent system built using the same method as the HTS entry to the Blizzard 2005 Challenge. The IBM systems (anonymized as System S) are explicitly labeled in the plots that follow.

4.1. English

For each of the Voices A and B, 620 sentences, broken down as follows, were synthesized and submitted to the organizers: 100 conversational sentences, 100 news sentences, 200 novel sentences, 200 SUSs and 20 emphasis sentences. The emphasis sentences were not used in this evaluation; subsets of sentences from the remaining categories were used to populate the 5 parts of the test described (Section 1 used sentences from the novel, conversational and news domains whereas each pair in Section 2 was taken fully from either the news or the novel domain).

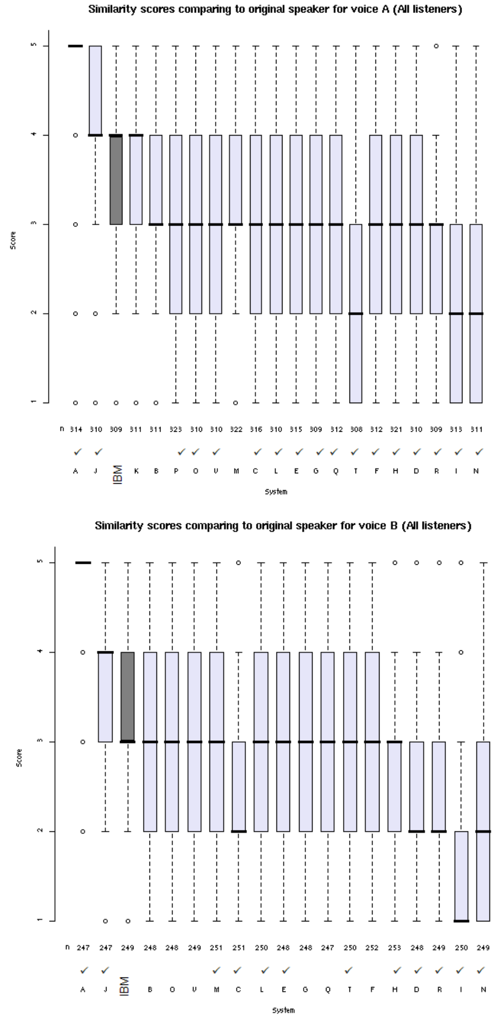

Figure 1: *Target similarity for English Voice A (top) and Voice (B) bottom*

Figure 1 shows the results of the speaker similarity task for the English voices A and B, and figure 2 shows the overall mean opinion scores for these two voices. We can make a few remarks based on these results: (i) the IBM English systems attain very good performance for the two voices, (ii) the MOS performance of the system degrades very gracefully when only a fraction of the data is used (3.7 for Voice A vs. 3.6 for Voice B; median of 4.0 in both cases), (iii) for the full voice, there seems to be a noticeable correlation between the similarity scores and the MOS across systems, suggesting that perhaps listeners are influenced by overall quality when judging similarity; this trend is less noticeable for Voice B, and it is less clear why there is a drop in the listeners' judgment of similarity for the IBM English B voice given the fairly steady MOS across voices already noted.

The SUS task was evaluated by asking listeners to transcribe what they heard. The word-error-rate (WER) between the synthesis texts and the transcriptions, subject to suitable normalizations, is plotted in figure

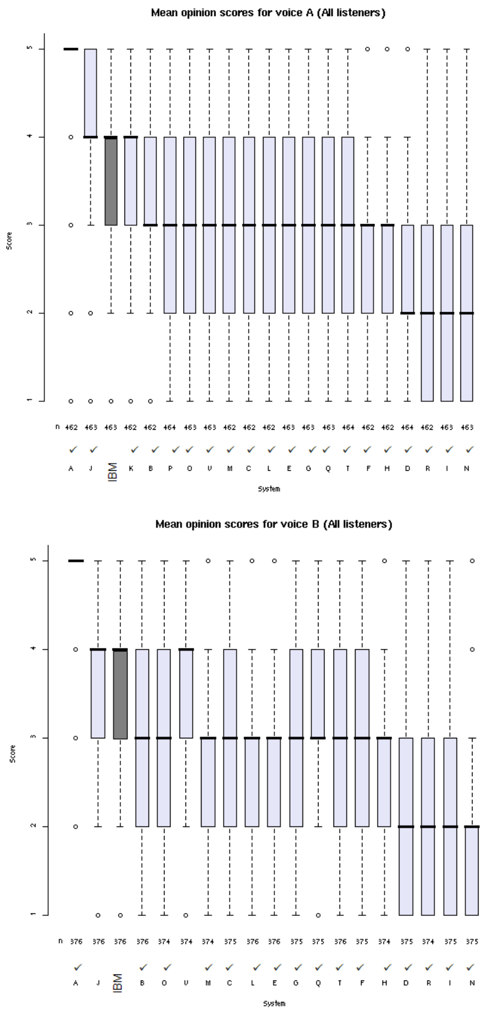

Figure 2: *Overall MOS for English voice A (top) and voice B (bottom)*

3 for the 2 English voices. Excepting the natural voice, only one system in each of the 2 voices attains significantly lower WER.

4.2. Mandarin

For Mandarin, 697 sentences were synthesized and submitted for evaluation: 647 news sentences and 50 SUS sentences.

Figures 4 and 5 show the results of the speaker similarity task and mean opinion score task respectively for the Mandarin voice. We can see from these plots that several systems, including ours, are clustered in terms of performance. In fact, only the natural voice receives a significantly higher score for these two tasks.

The results of the SUS Mandarin task is shown in figure 6 in terms of character-error-rate (CER) between the synthesis texts and the transcriptions, after proper normalizations. We also find for this task a cluster of top performers from which the IBM system does not perform significantly different; only the natural voice attains a lower error score. We also notice a similar behavior when, instead of computing character discrepancy, the error rate is quantified with Pinyin sequences with and without tones (not plotted).

5. Conclusions

In this paper we have presented the IBM systems submitted to the 2008 Blizzard speech synthesis challenge for UK English and Mandarin. We have provided algorithmic descriptions of the systems behind these submissions and described the evaluation tasks and performance attained by the IBM samples. This year's Blizzard installment afforded us a chance to work with rich and challenging datasets that motivated further algorithmic changes in our systems to address the peculiarities of the corpora. The result were systems that rose to the challenge by being among the top performers for both languages in the variety of evaluation tasks considered.

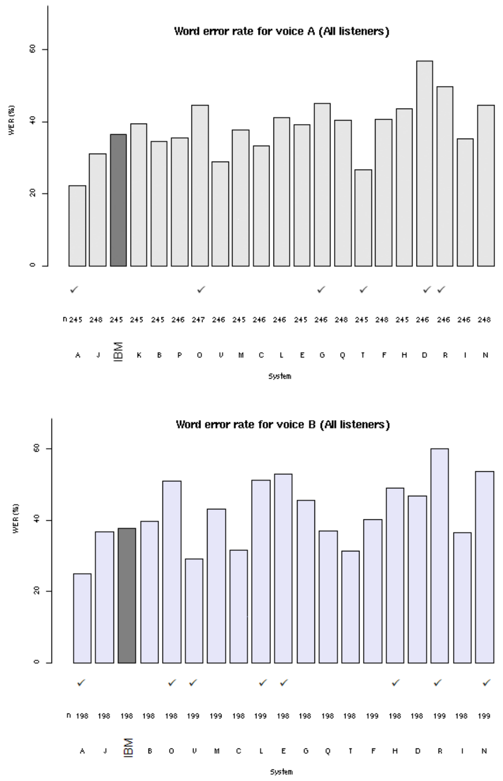

Figure 3: *WER for English voice A (top) and voice B (bottom)*

6. Acknowledgments

We would like to thank Andy Aaron for careful proofreading and making corrections to the training script, Larry Samsone for helping run some of the pre-evaluation listening tests, Ryuki Tachibana for sharing his alignment correction tools, and the Blizzard 2008 team for organizing the Blizzard Challenge and donating the resources to build the voices.

7. References

[1] J. Kominek and A. Black, "CMU ARCTIC databases for speech synthesis." Language Technologies Institute, Carnegie Mellon University, Tech. Rep., 2003, CMU-LTI-03-177 http://festvox.org/cmu_arctic.

[2] J. Pitrelli, R. Bakis, E. Eide, R. Fernandez, W. Hamza, and M. Picheny, "The IBM Expressive Text-to-Speech Synthesis System for American English." *IEEE Trans. Audio, Speech and Lang. Processing*, vol. 14, no. 4, pp. 1099–1108, July 2006.

[3] S. Shechtman, "Maximum-likelihood dynamic intonation model for concatenative text to speech system." in *Proc. Sixth ISCA Workshop on Speech Synthesis*, P. Wagner, J. Abresch, S. Breuer, and W. Hess, Eds. Bonn, Germany: Rheinische Friedrich-Wilhelms-Universität Bonn, August 2007, pp. 234–239.

[4] M. Zheng, Q. Shi, W. Zhang, and C. Lianhong, "Grapheme-to-Phoneme conversion based on TBL algorithm in mandarin TTS system," in *Proc. Interspeech*, Lisbon, Portugal, 2005, pp. 1897–1900.

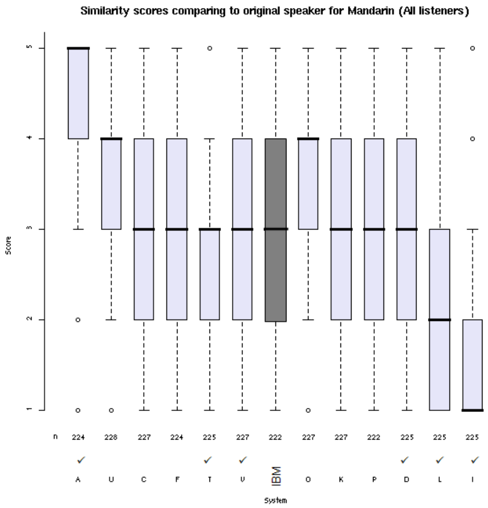

Figure 4: *Target similarity for Mandarin voice*

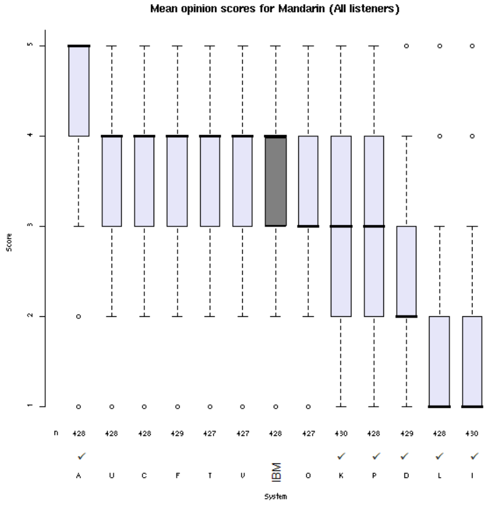

Figure 5: *Overall MOS for Mandarin voice*

[5] Q. Shi, X. Ma, W. Zhu, W. Zhang, and L. Shen, “Statistic prosody structure prediction,” in *Proc. IEEE TTS Workshop*, Santa Monica, U.S., 2002.

[6] W. Zhu, W. Zhang, Q. Shi, and F. Chen, “Corpus building for data-driven TTS systems,” in *Proc. IEEE TTS Workshop*, Santa Monica, U.S., 2002.

[7] R. A. J. Clark, M. Podsiadlo, M. Fraser, C. Mayo, and S. King, “Statistical analysis of the blizzard challenge 2007 listening test results.” in *Proc. Blizzard Challenge Workshop*, August 2007.

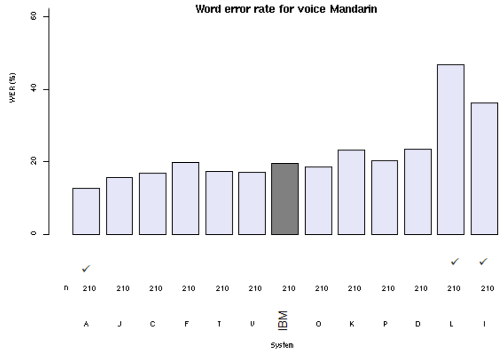

Figure 6: *WER for Mandarin voice*