The MARY TTS entry in the Blizzard Challenge 2008

Marc Schröder, Marcela Charfuelan, Sathish Pammi, Oytun Türk

DFKI GmbH, Language Technology Lab, Saarbrücken and Berlin, Germany

firstname.lastname@dfki.de

Abstract

The present paper reports on the DFKI entry to the Blizzard challenge 2008. The main difference of our system compared to last year is a new join model inspired by last year's iFlytek paper; the effect seems small, but measurable in the sense that it leads to the selection of longer chunks of consecutive units. In interpreting the results of the listening test, we correlate the ratings to various measures of the system. This allows us to explain at least some part of the variance in MOS ratings.

Index Terms: speech synthesis, unit selection, join costs

1. Introduction

For DFKI, the Blizzard challenge is a welcome annual joint activity, helping us to get a better understanding of the factors relevant for the perceived quality of synthetic speech. One relevant aspect is, of course, the "competition" aspect: we want to know how our system compares to the community at large. For us, however, a second aspect is becoming substantially more important: we want to understand *why* one approach is better than another one, and which objective measures can be used to predict the perceptual quality. This year's release of detailed ratings for individual synthesis results has allowed us to make some simple steps into this direction (see Section 4 below).

As in previous years, the challenge consisted in building voices from several collections of speech recordings. Whereas the previous Blizzard challenges worked with US English, this year's data was in British English and in Mandarin Chinese. DFKI participated only for the British English part.

We start by presenting a short summary of our system, with an emphasis on improvements since last year, before describing the process of building the Blizzard voices and presenting and discussing the results of the listening test.

2. The MARY system

The current architecture of the open source MARY (Modular Architecture for Research on speech sYnthesis) platform is shown in Figure 1. MARY is a stable Java server capable of multi-threaded handling of multiple client requests in parallel. The design is highly modular: a set of configuration files, read at system startup, define the processing components to use. For example, the file `german.config` defines the German processing modules, while `english.config` defines the (US) English modules. If both files are present in the configuration directory, both subsystems are loaded when starting the server. Each synthesis voice is defined by a configuration file: `german-bits1.config` loads the unit selection voice bits1, `german-hmm-bits1.config` loads the HMM-based voice hmm-bits1, etc. More details on the MARY architecture can be found in [1].

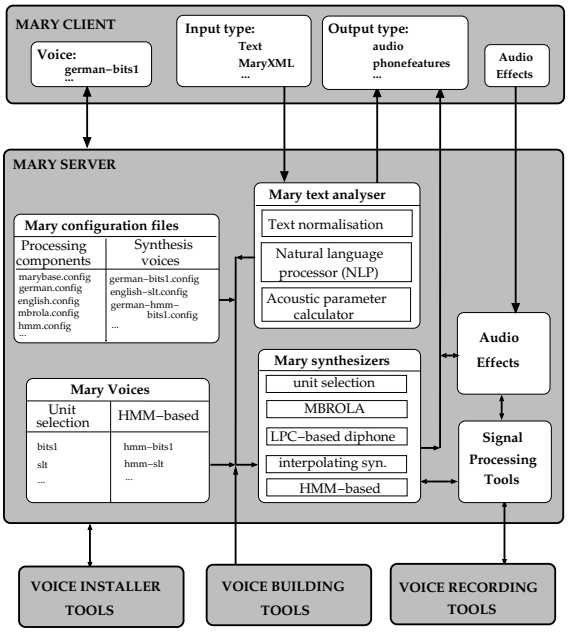

Figure 1: Mary TTS platform release version 3.6.0.

Currently, the list of available waveform synthesisers includes a unit selection synthesiser [2], an MBROLA diphone synthesiser, an LPC-based diphone synthesiser provided by FreeTTS, an experimental interpolating synthesiser [3] and a new HMM-based synthesiser ported to Java from the excellent HMM-based synthesis code from the HTS project (http://hts.sp.nitech.ac.jp/). The MARY text analyser components are described in [4]. The audio effects component is a new component designed to apply different effects on the audio produced by the different synthesisers. The effects are set through the audio effects GUI of the MARY client component. The Voice installer tools component is used for downloading and installing new voices or removing already installed ones. The voice recording tool is a new component designed to facilitate the creation of speech synthesis databases. The voice building tools component has been improved with respect to the previous version, as will be described below.

2.1. Improvements in voice building procedure

As compared to our participation in last year's Blizzard Challenge [1], we have significantly improved our voice building tool to create new synthetic voices [5]. The latest version of the open source MARY TTS includes not only the necessary components for unit selection voice creation but also components for HMM-based voice creation.

2.1.1. Groups of components

The voice building tool that is currently available includes the following groups of components:

- `Raw acoustics`, including tools for extracting pitch marks and MFCC coefficients.
- `Feature computation tools` predict linguistic features using the MARY system.
- `Labelling components` include automatic labelling tools like EHMM [6], various alignment steps to make sure the features and the labels are in synchrony, and a quality control component.
- `Acoustic models` train models predicting target acoustics.
- `Unit selection files`, generating data files for efficient use in the run-time unit selection system, and training the CART for pre-selection of candidate units.
- `HMM-based files`, generating data files required to train models and generating the files required for the run-time HMM based voice.
- `Install voice` copies the resulting files to their target location in the run-time MARY system.

Several of the voice import components call external components, for example: Wagon for CART training, Praat for pitchmarking, EHMM for labelling, HTS/HTK for HMM model training etc. Many variables of the component's processing can be configured through the GUI, but they are intended to produce meaningful behaviour with the default settings.

2.1.2. Automatic Labelling

The voice building tool supports two automatic labelling tools: EHMM and Sphinx [7]. In the Sphinx-based labeller, semi-continuous HMMs are used, a skip state is allowed, 13 mel-frequency cepstral coefficients (MFCC), their delta and delta-delta coefficients are used and context-dependent models are trained for performing forced-alignment [8] in the decoding phase. In the EHMM tool, continuous models with one Gaussian per state, left-to-right models with no skip state and context-independent models trained with 13 MFCCs are used to get force-aligned labels. Normally we use the EHMM tool because it is well tuned to automatic labelling for building synthetic voices. Its main features are:

- It can initialise with flat start initialisation as well as model initialisation. The latter makes it possible to perform model adaptation.
- It supports modelling of short, long, and optional pauses, which produces better alignment of speech segments.
- It uses a context-independent acoustic model as context dependent models tend to blur the label boundaries.
- More resolution can be supported with a frame shift of 5 milliseconds resulting in sharper boundaries.

2.1.3. Creation of HMM-based voices

For creating HMM-based voices we use a version of the speaker dependent (or adaptive) training scripts provided by HTS [9], adapted to the MARY platform. The scripts and programs used for training HMM voices for the MARY platform have been slightly modified from the original HTS scripts, basically they have been changed to use context features predicted by the MARY text analyser instead of the Festival one. We provide a patch file to be applied to the HTS training scripts so the MARY voice building tools can be used in a straightforward way. The main changes included in this patch are:

- German or English language as parameter.
- Calculate bandpass voicing strengths for mixed excitation.
- Composing training data from mel-generalised cepstrum (mgc), log F0 and strength files.
- Extracting monophone and fullcontext labels from MARY context features.
- The HMMs training script has been modified to consider bandpass voicing strengths for mixed excitation as additional parameters.

The current procedure for creating a new HMM-based voice can be summarised in three steps: data preparation, training of HMM models and installation of a new voice into the MARY system, more details on this procedure can be found in [10].

2.2. Join Model

As the MARY system supports both unit selection and HMM-based voices, we are curious about methods for combining the best of both worlds. In this context, the paper by Ling et al. [11], presented at the Blizzard challenge 2007, has been very inspiring. In our submission, we experiment with one part of their approach, namely a statistically trained join model.

Our Blizzard 2008 system, as well as our Blizzard 2007 system [1], is a unit selection system selecting diphone units based on a combination of target costs and join costs. For each target diphone, a set of candidate units is selected by separately retrieving candidates from each halfphone through a decision tree, and retaining only those that are part of the required diphone. When no suitable diphone can be found, the system falls back to halfphone units. The most suitable candidate chain is obtained through dynamic programming, minimising a weighted sum of target costs and join costs. Our target cost function has a linguistic and an acoustic component. Linuistic target cost covers the linguistic properties of units and their suitability for a linguistically defined target. Acoustic target costs are used to compare a unit's duration and F0 to the ones predicted for the target utterance by means of regression trees trained on the voice data.

In the Blizzard 2008 system we use a different way of calculating join costs. In the previous system, the join costs were computed as absolute distance of mel-cepstrum and F0 parameters at candidate boundaries. In the current system, the join model is a context clustering decision tree, trained to model the transitions of acoustic features at halfphone boundaries. Similar to the iFlytek concatenation model [11], our join model uses as acoustic features the differential of mel-cepstrum and F0 between the first frame of the current halfphone and the last frame of previous halfphone. Unlike the iFlytek concatenation model,

we use halfphones instead of phones, and our acoustic features are extracted pitch-synchronously.

The JoinModeller component of the voice import tools in MARY builds a join model as follows: join cost features (differential of mel-cepstrum and F0 of all pairs of adjacent units) are calculated for all the labelled context dependent halfphone units in the database. Statistics across all the observations that have the exact same full-context model name are calculated. Most of the time there will be just one observation of each model name, therefore model clustering is necessary when creating the decision tree.

During unit selection synthesis, the join model, tree and PDF files are loaded in the MARY system and used to calculate a join cost. Given a target feature vector and a set of units to concatenate, a join cost is calculated as follows: a difference between the last frame of a unit and the first frame of the next unit is calculated. Each difference is weighted by the likelihood of the difference under the join model, that is, the target feature vector is looked up in the join tree, a mean and variance is retrieved and a Mahalanobis distance is calculated between the mean and the difference of units.

3. Building the Roger voice

For the first time, the Blizzard challenge consisted in creating voices for languages other than US English: British English and Mandarin Chinese. As there is not yet a toolkit for quickly supporting new languages in the MARY platform, we did not attempt to build a Mandarin voice; for the British English voices, we "tricked" our system into believing it was dealing with US English, and merely provided a different pronounciation lexicon for these voices. This ad hoc approach is now being replaced with a clean framework for supporting new languages and country-specific variants of a language, so that it will be easier for us to participate in future Blizzard challenges involving new languages.

The speech material from the British English speaker "Roger", provided by CSTR Edinburgh, consists of 9509 utterances with a total of 15 hours of speech. It includes as a subset the traditional, phonetically balanced "*arctic*" subset, of 1132 utterances corresponding to 87 minutes of speech.

The speech material consists of several subsets designed to allow for the study of emphasis while assuring prosodic and diphone coverage [12]. The following subsets are included:

- *news*, read in a matter-of-fact and rather fast speaking style;
- *carroll*, material from Lewis Carroll's children stories, read in a lively and spirited manner, i.e. with large prosodic variations;
- *unilex*, material consisting of single words produced with a range of intonations, to assure prosodic and phonetic coverage in phrase-final position;
- *emphasis*, material embedding emphasised proper names into carrier sentences to ensure diphone coverage on emphasised speech;
- and two small specialised sections for *spelling* and *address* reading.

Two voices were to be built: voice A from the full set of data, and voice B from the arctic subset.

We used the *unisyn* lexicon kindly provided by CSTR to build our lexicon. We followed the documentation to create an RP lexicon. Entries not found in the training set were added manually into a user dictionary.

During the voice building process, only a very small number of the utterances were automatically discarded (36 for voice A, 3 for voice B). In particular, this means that we did use the subsets of the data that do contain emphasised speech (carroll, unilex, emphasis, spelling) alongside the subsets that do not (arctic and news), even though our NLP components do not model emphasis. This can be expected to result in uncontrolled variation, as our target costs do not contain any "emphasis" features. Notably, our acoustic target models cannot model the acoustic correlates of emphasised speech due to the lack of a suitable predictor feature.

According to the clarified rules of the Blizzard challenge 2008, all processing was done separately for voice A and voice B, so that no data from voice A whatsoever was used to build voice B. In particular, we automatically force-aligned the speech data from our custom segment predictions using EHMM, and trained F0 and duration target models as regression trees, separately for voice A and voice B.

For the new join model, the idea was to use statistical criterions in the calculation of join costs, similar to the system reported in [11]. This approach was appealing because it make use of conventional unit selection methods and new HMM-based methods, which now are also available in the MARY system. For our new join model we have created a context clustering decision tree in HTS format, the same format of the trees used in the HMM-voices in the MARY system.

The method for creating this tree is similar to the way the duration model is createad in HTS HMM-based voices. First of all we have calculated statistics: means, (diagonal) covariances and number of repetitions of a unit, across the join cost features (differential of mel-cepstrum and F0) of all pairs of adjacent context-dependent halfphone units in the database. The means and covariances were used to create single-Gaussian single-state HMM models, in HTK format, for all the units in the database. The number of repetitions were used to create a HTK stats file. We have also created a set of linguistic questions for decision tree-based context clustering based on the target features used in the system. These three elements, HMM models, stats file and questions were used together with the HTS-HTK[1] HHEd command to create a context clustering decision tree. This tree was converted to HTS format using another HHEd command. All this procedure has been included in the JoinModeller tool of the MARY voice import tools.

Initially, the run-time component for computing the new join cost was rather slow, requiring $\sim$10 seconds to synthesize one utterance, compared to $\sim$100 ms using the old join cost. We traced the problem to the string-based traversal of HTS trees in our code; once we replaced this with direct comparison of feature values encoded as bytes, the code became nearly as fast as the old one.

We built the classification trees for pre-selecting candidate units as we did last year, through a combination of a user-defined top-level tree and automatically grown sub-trees from top-level leaves using wagon and an acoustic distance criterion. At this step, we unfortunately overlooked a hard-coded maximum size of top-level leaves, which would discard any units in top-level leaves beyond a maximum value. This had been introduced in earlier times because we had observed that wagon is using much more time when building trees from larger collec-

[1] We have used the HTK version 3.4. patched with the HMM-based Speech Synthesis System (HTS) version 2.0.1

	MOS		WER (native)	
	Voice A	Voice B	Voice A	Voice B
DFKI	2.8	3.2	0.26	0.21
All systems	2.9	2.8	0.25	0.28

Table 1: Mean Opinion Score (higher values are better) and Word Error Rate (lower values are better) for the DFKI entry compared to the average of all systems (excluding natural speech)

tions of data. The oversight was detected only after the listening tests had been completed. It hit our voice A much more severely than voice B: out of a total of 78,500 halfphone units in voice B, the tree contained 72,000, i.e. more than 90%; for voice A, however, from the total of 819,000 halfphone units, only 247,000 or 30% were actually contained in the tree. This means that the speech synthesised with our voice A actually used only 30% of the available speech material, while the acoustic target models for voice A were trained on the full data set.

If competition was the main aspect of the Blizzard challenge, this oversight in our voice creation process would certainly be highly regrettable; however, as we are interested in understanding *reasons* for good or bad performance, it can actually be considered an opportunity to investigate the effect that using only part of a huge database has on synthesis quality.

4. Results and discussion

The DFKI entry in the Blizzard listening tests is identified by the letter Q. Average Mean Opinion Score (MOS) and Word Error Rate (WER) values are reported in Table 1. It can be seen that the DFKI entry is slightly below the average of all systems for voice A, and better than the average for voice B.

4.1. Analysis of voice A

The MOS for DFKI's voice A is substantially *worse* than for voice B. A priori, this is against expectations, and it is also against the trend for unit selection systems (e.g., the Festival reference system, system B). Given that with more units, the coverage can be expected to be better, the expectation would be that for voice A, it is more likely to find suitable units for a given target, and that less severe discontinuities occur at join points, because either longer chunks can be selected from the speech database, or candidates can be selected that fit better to their context.

An obvious hypothesis is that the error in building the pre-selection tree, leading us to ignore 70% of the speech material for voice A, is a cause for the worse scores. However, the sheer *amount* of data cannot be the sole cause: our voice A still contains nearly four times the amount of speech data in voice B. It is therefore worth taking a closer look.

Figure 2 shows the distribution of diphones belonging to different styles, as found in the full database for voice A and accessible in our pre-selection CART. It can be seen that the distribution is drastically different. Whereas in the full database, styles with unemphasised speech (news and arctic) provide more than half of the diphones, less than a quarter of the diphones accessible via the CART are from these styles. Instead, 60% of the diphones in the CART are from the *carroll* set, children stories.

This finding seems to explain to a large extent the worse performance of our voice A over voice B, for two reasons:

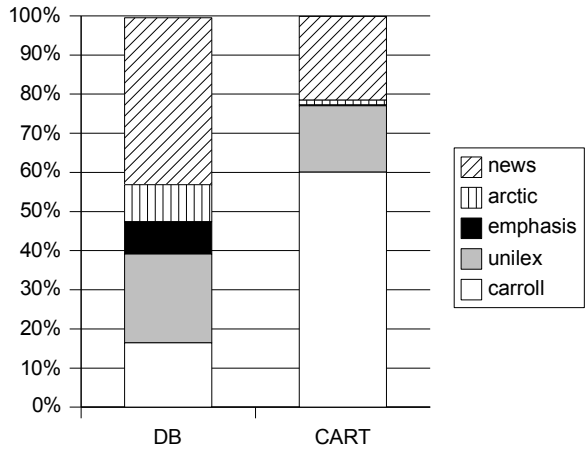

Figure 2: Relative number of diphones in different styles, for the full voice A database (left) and included in the pre-selection CART for voice A (right). Styles *spelling* and *address*, which together make up less than 0.5%, have been omitted for clarity.

Voice A		Voice B	
DB	CART	DB	CART
1779	1468	1538	1533

Table 2: Diphone coverage for our voices A and B; DB: diphones contained in the database; CART: diphones available for synthesis through the classification tree used for pre-selection of candidate units.

- on the one hand, the expressive material from the *carroll* and *unilex* sets is not optimally suited for the news and novel material used in the listening test;
- on the other hand, the acoustic target models trained on the full database do not match well the speech material available after pre-selection.

This issue does not affect our voice B, since it contains only a single style.

Furthermore, the diphone coverage of our voice A falls below that of our voice B, as can be seen in Table 2.

In summary, it seems plausible to conclude that the main reason for the worse performance of our voice A is the combination of suboptimal speech material, lower diphone coverage and the mismatch between acoustic target models and diphone candidates.

4.2. Relating objective measures to listener ratings

The organisers of the Blizzard challenge 2008 have released the rating scores for all individual utterances. This allows us to attempt relating them to objective measures, in order to better understand which objective measures, if any, are useful as predictors of subjective quality. We do this in a very simple way by computing correlations between average listener ratings per sentence and each of a range of measures. We compute correlations separately for voice A and voice B in order to be sure not to mix measures that are systematically different between the two voices.

One simple measure that could be expected to be informa-

Correlation MOS with	Voice A	Voice B
# diphone candidates	0.05	0.21
chunk length	0.28	0.42
cost best path	-0.37	-0.51

Table 3: Correlations between MOS ratings, averaged across listeners individually for every utterance in sections 3 and 4 of the listening test, and several objective measures. # diphone candidates: average number of candidates for every diphone in the Viterbi search; chunk length: average length of consecutive stretches in the selected path, measured in halfphones; cost best path: the total cost of the best path divided by the number of units in the path.

tive, based on the notion of diphone coverage, is the average number of candidates for every target diphone available in the dynamic programming. If more candidates often lead to a better synthesis result, there should be a positive correlation between the average number of candidates used for an utterance and that utterance's MOS score. However, this is not the case: correlations are tiny (Table 3). More fine-grained measures of coverage may be more suitable, such as the number of missing diphones (for which MARY falls back to halfphones), or the minimum number of candidates available for a diphone in the sentence. These will be computed in a future experiment.

Another simple objective and system-independent measure, which is often considered to be relevant for synthesis quality (e.g., [13]), is the length of consecutive chunks, i.e. the length of adjacent speech material selected from the database. The reasoning here is that with fewer concatenation points, there are fewer points where discontinuities can occur. Indeed, we find a weak correlation between average MOS score and average chunk length (see Table 3). However, this factor seems to be just one of many, as the proportion of variance in MOS explained by chunk length (as measured by r^2, the correlation squared) is only 8% (voice A) or 18% (voice B). In the scatterplot for voice B (Figure 3), it can be seen that much variation remains which appears to be unrelated to chunk length.

Within a given system, the most informative measure about the quality is the total cost, i.e. the weighted sum of target and join costs. Indeed, the purpose of this cost measure is exactly to approximate listener ratings as well as possible. Hence, for a given utterance, the average cost per unit in the best path should be a global measure for the utterance quality. If the target and join costs used were truly appropriate approximations of perceived synthesis quality, the correlation of average cost with MOS ratings should be very strong, i.e. close to -1 (obviously the correlation is expected to be negative, i.e. lower cost should correspond to higher ratings). In our case (Table 3), we are far from an ideal correlation, but the correlation is stronger than the other ones measured so far, explaining 14% of the variance in MOS ratings for voice A and 26% for voice B. Figure 4 shows the scatterplot and trendline for voice B.

While being the strongest correlation we found so far, cost is not an optimal measure for comparing the performance of systems and voices because the cost is highly system-specific, and even within a system, it may not be appropriate to compare the absolute values of the cost function across voices. Therefore, it would be preferable to find more primitive measures that correlate with MOS ratings, which in turn could be used to define a revised cost function.

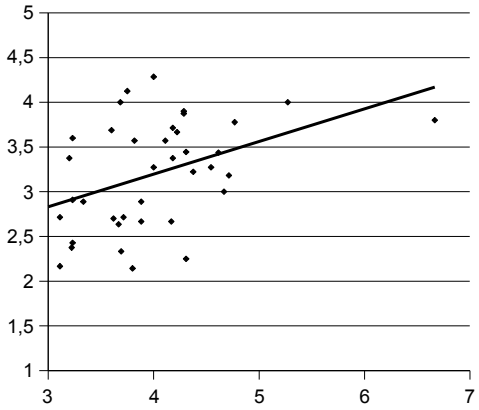

Figure 3: Scatterplot of MOS ratings over average length of selected chunks in halfphones, for DFKI voice B.

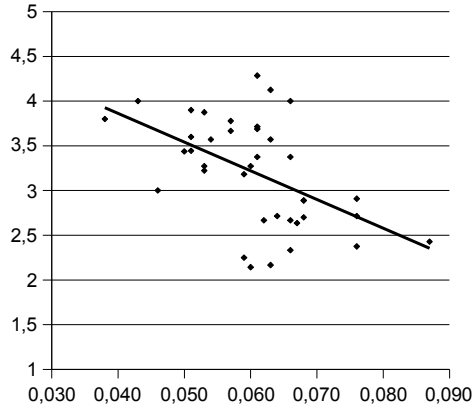

Figure 4: Scatterplot of MOS ratings over average unit cost in best path, for DFKI voice B.

4.3. The new join model

The DFKI voices submitted to the Blizzard challenge used the new join model as described above. We also built variants of these voices that used our previous join cost measure, a simple absolute distance for F0 and mel cepstrum. There were only minor differences between the two versions. Informal listening tests with a set of test sentences from the novel and news domain yielded no systematic differences. However, it could be observed that the lengh of consecutive chunks selected using the new join model was on average slightly longer (by 0.2 – 0.3 halfphones on average) compared to the versions with the old join cost. As shown above, this may result in a slight improvement in MOS scores, even if the effect may have been too subtle to be easily perceived in an informal listening test.

5. Conclusion

This paper has described the process of building the DFKI entry to Blizzard 2008, the results of the listening test, and notably an attempt to relate the listener ratings to objective measures. One lesson learnt through this year's participation is that increasing the amount of data as such does not automatically improve quality, nor does the average number of diphone candidates in the selection process predict perceived quality. We did find, however, some limited correlations with MOS for the length of selected speech chunks, as well as for our cost measure which combines linguistic and acoustic target costs with an acoustically trained join model.

The joint analysis of objective measures and MOS scores that we have started to explore in this paper seems promising to us. In the future, we will carry out a much broader analysis, generating a selection of measures during the synthesis process in order to relate them to MOS scores later. Hopefully that approach will help deepen our understanding of the objectively measurable foundations of perceived synthesis quality.

6. Acknowledgements

The research leading to these results has received funding from the DFG project PAVOQUE and from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement no. 211486 (SEMAINE). The authors would like to thank Zhen-Hua Ling from the University of Science and Technology of China for his helpful explanation of the iFlytek concatenation model. We are grateful to the Festival, Sphinx, Praat, Snack, HTS and HTK development teams for making their software open source and publicly available.

7. References

[1] M. Schröder and A. Hunecke, "Mary TTS participation in the blizzard challenge 2007," in *Proc. Blizzard Challenge 2007*, Bonn, Germany, 2007.

[2] ——, "Creating German unit selection voices for the MARY TTS platform from the BITS corpora," in *Proc. Sixth ISCA Workshop on Speech Synthesis*, Bonn, Germany, 2006.

[3] O. Turk, M. Schröder, B. Bozkurt, and L. Arslan, "Voice quality interpolation for emotional text-to-speech synthesis," in *Proc. Interspeech 2005*, Lisbon, Portugal, 2005.

[4] M. Schröder and J. Trouvain, "The German text-to-speech synthesis system MARY: A tool for research, development and teaching," *International Journal of Speech technology*, vol. 6, pp. 365–377, 2003.

[5] S. Pammi, "Voice import tools tutorial: How to build a new voice with voice import tools," http://mary.opendfki.de/wiki/VoiceImportToolsTutorial, 2008.

[6] K. Prahallad, A. Black, and R. Mosur, "Sub-phonetic modeling for capturing pronunciation variations for conversational speech synthesis," in *Proc. ICASSP 2006*, Toulouse, France, 2006.

[7] The CMU Sphinx Group, "Sphinx: Open source speech recognition engine," http://cmusphinx.sourceforge.net/html/cmusphinx.php, 2008.

[8] J. Kominek and A. Black, "A Family-of-models approach to HMM-based segmentation for unit selection speech synthesis," in *Proc. Interspeech 2004*, Jeju Island, Korea, 2004.

[9] K. Tokuda, H. Zen, J. Yamagishi, A. W. Black, T. Masuko, S. Sako, T. Toda, T. Nose, K. Oura, and others., "HMM-based speech synthesis system (HTS): Speaker dependent training demo," http://hts.sp.nitech.ac.jp/, 2008.

[10] M. Charfuelan, "Voice import tools tutorial : How to build a HMM-based voice for the MARY platform," http://mary.opendfki.de/wiki/HMMVoiceCreation, 2008.

[11] Z. Ling, L. Qin, H. Lu, Y. Gao, L. Dai, R. Wang, Y. Jiang, Z. Zhao, Y. Yang, J. Chen, and G. Hu, "The USTC and iFlytek speech synthesis systems for blizzard challenge 2007," in *Proc. Blizzard Challenge 2007*, Bonn, Germany, 2007.

[12] V. Strom, A. Nenkova, R. Clark, Y. Vazquez-Alvarez, J. Brenier, S. King, and D. Jurafsky, "Modelling prominence and emphasis improves unit-selection synthesis," in *Proc. Interspeech 2007*, Antwerp, Belgium, 2007.

[13] A. Schweitzer, N. Braunschweiler, T. Klankert, B. Möbius, and B. Säuberlich, "Restricted unlimited domain synthesis," in *Proc. Eurospeech 2003*, Geneva, Switzerland, 2003.