SYNTHESIZER VOICE QUALITY OF NEW LANGUAGES CALIBRATED WITH MEAN MEL CEPSTRAL DISTORTION

John Kominek, Tanja Schultz, Alan W Black
jkominek@cs.cmu.edu, awb@cs.cmu.edu

Language Technologies Institute, Carnegie Mellon University, USA

ABSTRACT

When developing synthesizers for new languages one must select a phoneset, record phonetically balanced sentences, build up a pronunciation lexicon, and evaluate the results. An objective measure of voice quality can be very useful, provided it is calibrated across multiple speakers, languages, and databases. As a substitute for full listening tests, this paper adopts mean mel-cepstral distortion as a measure of spectral accuracy, and proposes systematic variation of a known English corpus as a method of calibration. We find that doubling the database size reduces MCD by 0.12, while reverting to a grapheme-based voice increases it by 0.27. This offers a frame of reference for estimating voice quality, which is applied to a test suite of 8 non-English languages.

1. INTRODUCTION

Our interest lies in extending the reach of speech synthesis to "new," i.e. previously uncovered languages, by making the voice development task easier for non-specialists. The open source Festival/Festvox speech synthesis toolkit [1] contains working examples of major languages such as English and Spanish, plus templates for less-resourced languages. Dozens of synthesizers have been built for the Festival system. Nonetheless, these tools are not for the faint of heart – they require a software developer having considerable expertise with speech and language technologies, plus the patience to manually review labeled databases. In addition, the developer needs frequent access to a native speaker in the target language. The combination of language and technology expertise conspires to form a bottleneck to progress.

We are attempting to relieve this bottleneck by creating a a simplified development framework that is an enhancement of Festvox, and is part of a larger project called SPICE. The SPICE project aims to deploy a web-based toolkit for the rapid development of ASR and TTS technologies [2]. The first version of this toolkit has been now running as a live server for one year [3], and has been used in laboratory courses taught at Carnegie Mellon University in the U.S. and at Karlsruhle University in Germany. The TTS component is more advanced in the sense that it guides the user through the various stages of text and speech collection, phoneset definition, pronunciation dictionary development, voice creation, and evaluation. As an additional design goal, we want the system to produce an intelligible (if not high quality) voice with as little as 5-10 minutes of recorded speech, and to noticeably improve with additional data.

The efforts of a dozen students to create new speech synthesizers with the SPICE toolkit are described in [4]. We found that 5 minutes of speech (i.e. the reading of about 100 short sentences) is enough for an initial voice – with 15 minutes preferred – provided the material is phonetically balanced and the accompanying pronunciation dictionary is complete and accurate. Yet anecdotal experience confirms that lexicon creation is labor intensive, fatiguing, and error prone. While 500 lexical entries, for example, is small compared to 100,000 typical of full-language coverage, motivating users to complete just that much is not easy. One student expressed the general sentiment as "can't you just ask me about words that it can't predict correctly?".[1]

There are a number of ways to address this concern, including the simple retort "no." More optimistically, one can refine the graphical user interface used for lexicon creation, with the intention of reducing human burden [5]. Also, while it introduces issues of consistency, multiple native speakers can work on the lexicon, if available. Yet, none of these suggestions offers an order-of-magnitude reduction in effort that our users implicitly seek. In response one can switch from phoneme-based to grapheme-bases synthesis, as such a synthesizer requires no pronunciation dictionary. This is inherently limited, for no human language possesses a one-to-one mapping between graphemes and phonemes. Less drastic is to retain phoneme-based synthesis, but to economize on human effort by seeking the minimal amount of information (i.e. word pronunciations) that is needed.

Achieving economy of effort presupposes a way to self-measure a system's knowledge as it is being built. We want to operationalize this at two levels: 1) measuring the correctness of a particular lexical entry, that it may "just ask

1 The "it" refers to the system's incrementally learned letter-to-sound rules, used to predict the pronunciation of the next word presented to the user.

about the words it doesn't know", and 2) measuring the voice quality as a whole, i.e. knowing when it is "good enough." This paper focuses on the second aspect.

We adopt mean mel-cepstral distortion (MCD) as an objective measure of global voice quality. To provide calibration marks we have run extensive tests on English, against which builds of other languages may be compared. Our complement of languages in this study are French, German, Tamil, Hindi, Mandarin, Bulgarian, Thai, and Konkani. Konkani is minor language of central India notable for lacking a native writing system. It is without question the most "under-resourced" language we have encountered.

Section 2 describes our testing framework used to address a list of specific questions (see 2.4) that focus on automatic evaluation of TTS for new languages. Section 3 provides results that suggest answers. Section 4 summarizes and points out unfinished business.

2. MEASUREMENT AND TESTING

Before defining our formulation of the MCD objective measure, there are a number of caveats worth stating at the outset. First is that it is not by any means complete, nor fully reliable. That is, there are many other factors that contribute to the perception of voice quality. For example – this is point number two – it takes no account of speech dynamics, either short-range differentials or long-range prosodic effects. Thirdly, distortions in the pitch contour are ignored. Thus, for tonal languages such as Thai, Cantonese, and Mandarin, the measurements must be considered suspect. Nonetheless, [6] shows that when this distortion measure decreases the corresponding voice quality *does* improve, so, in lieu of something better, it is a useful proxy.

2.1. Mel-cepstral distortion as an objective measure

In speech processing systems it is common to analyze a waveform into a sequence of multi-dimensional coefficients (vectors) at regularly spaced intervals called frames. For TTS applications, typical parameters are 25-D mel frequency-scaled cepstral coefficients with a frame step size of 5 ms. We represent this sequence of frames as $v_d(t)$ where d is the dimension index ranging from 0..24, and t is time, or more precisely, the frame index. Two waveforms – the target v^{targ} and reference v^{ref} – have a mean mel-cepstral distortion defined as an extension of the simple Euclidean norm, such that

$$MCD(v^{targ}, v^{ref}) = \frac{\alpha}{T'} \sum_{\substack{t=0 \\ ph(t) \notin SIL}}^{T-1} \sqrt{\sum_{d=s}^{D} (v_d^{targ}(t) - v_d^{ref}(t))^2} \quad (1a)$$

$$\alpha = \frac{10\sqrt{2}}{\ln 10} = 6.14185 \quad (1b)$$

where the scaling factor α is present for historical reasons [6], the shorter of the two wavefiles is given as $T = min(|v^{targ}|, |v^{ref}|)$ frames in length such that $T' \le T$ is the number of non-silence frames, while the expression $ph(t) \notin SIL$ excludes frames that lie inside silence regions, and s is the "starting" dimension of the inner sum, and equals either 0 or 1. When s = 0, eqn (1) includes the *zeroth* cepstral dimension, the component known to correspond to overall signal power. In this paper, results are computed with s = 1, i.e. the power term is ignored. We adopt this choice so that the distortion measure is not influenced by the speaker's loudness, something that is less controlled in web-based audio collection compared to studio recordings. Note that some numbers reported in the literature implicitly assign s = 0, and thus are not directly comparable to ours.

For our purposes the target and reference waveforms are the synthesized and original versions of the same utterance. Synthesizing from text, though, presents a problem. Because the durations of phonemes are *predicted,* the time alignment will diverge from the original, rendering eqn (1) misleading. This can be improved by performing dynamic time warping between the target and reference, but such compensation is only partially reliable. Instead, it is better to use the original waveform as a detailed template for resynthesis. The reference and target frames thus align 1-to-1, improving the precision of distortion measurements.

2.2. Test suites with 10-fold cross validation

When the test set consists of multiple utterances (as is normal), the individual MCD scores of wavefile pairs are averaged. In addition, we perform 10-fold cross validation on a given database using a 90/10 training/test split. So if a database has 1000 utterances, these are divided into 10 partitions p, each with a training set of 900 and a test set of 100. The test utterances constitute ten non-overlapping sets and are uniformly sampled at indices $\{n\}_p$ given by the expression $\{n\}_p = (n + p) \, mod \, 10 = 0, \; p = 0..9$. Having ten values per experiment makes it is easy to compute standard deviations for the purpose of significance testing.

2.3. Prediction features

When performing resynthesis, the task of the synthesizer is to compute the target from a combination of the reference and the original text: $v^{targ} = F(text, v^{ref})$ with the constraint that attributes of the reference such as phoneme durations may be a part of the function, but not the mel-cepstral vectors themselves. (That wouldn't be fair game.) The choice of prediction features and the mathematical machinery used for constructing the approximation function F is a critical element of synthesizer design. Festival uses CART trees for predicting values from feature vectors. The machine learning algorithm used for training CART trees is

(in a small play on words) a program named *wagon*. *Wagon* is a part of the publicly available Edinburgh Speech Toolkit [7], and is integrated into the CLUSTERGEN [8] component of Festival used for these experiments.

Let a feature vector $\boldsymbol{x}$ have dimension N. For a frame at time index t, let $\boldsymbol{x}(t)=(f_1(t), f_2(t), \ldots, f_N(t))$ where each vector component has a corresponding function f that defines the feature value. As an example, one function may define the name of the phoneme to which the current frame belongs. Next we introduce a set of binary-branching CART trees with the function $\Gamma_{ph}(\boldsymbol{x})$ such that each phoneme-state ph has a dedicated tree. In combination, these trees (typically containing thousands of nodes) predict the melcep vectors for the target waveform from feature vectors.[2] Each frame is predicted separately in sequential order.

$$\boldsymbol{X}^N \rightarrow \mathbb{R}^D: \quad v^{targ}(t)=\Gamma_{ph(t)}(\boldsymbol{x}(t)) \tag{2}$$

We divide features into four classes: the set of name symbolics, position values, IPA symbolics, and linguistic symbolics. Name symbolics include the name of the current phoneme and that of the immediate neighborhood, plus the names of HMM states within a phoneme. Position values include the location of a frame within a state, e.g. how far it is from the starting time, plus derivative values. IPA symbolics are a subset of International Phonetic Association-defined features. For example manner and place of articulation. These depend on and are derived from the phoneme set of a given language. Linguistic symbolics are the output of high-level functions such as parts-of-speech taggers and functions that decompose words into syllables.

This classification is not arbitrary. Name symbolics requires only that each basic speech unit has a unique name. Names may equate with the classical concept of phonemes, or may be graphemes, or some other symbol set. Position values also share the property of being language-independent, but are different in being real-valued numbers. IPA symbolic features, in contrast, are language-dependent and assume that a) the names are phonemes, and b) the phoneme set is appropriate. This issue is highly relevant because the ultimate target users of SPICE are people without deep background in phonetics. Finally, the demands presented by linguistic features is even higher: namely, a computational linguist that can program functions in Lisp.

Being able to demonstrate that language-dependent features are not critical to voice quality would prove to be a great relief. Table 1 is a prelude to the results of Section 3, and summarizes results that demonstrate exactly this.

2 It is not uncommon to refer to melceps as "feature vectors." This nomenclature should not be confused with the present usage. CART trees transform points of one feature space to another.

Feature class	*Number*	*Lang-dep.*	*Δ MCD*
no CART trees	1	no	baseline
name symbolics	16	no	- 0.452
position values	7	no	- 0.402
IPA symbolics	72	yes	- 0.001
linguistic sym.	14	yes	+ 0.004

Table 1. Four features classes, number of features in each class, language-dependent status, and change of MCD. Absolute changes below 0.08 are not statistically significant.

2.4. Investigative questions

Within our framework we pose the following questions.

- How important are the language-dependent features in CART tree training?
- What minimal leaf node size ("stop value") is optimal?
- At what rate does a voice improve as more speech is collected? Is there a threshold beyond which collecting additional speech offers little added benefit?
- At what rate does a voice improve as the phoneme-based pronunciation lexicon is expanded? Is is more important to work on the lexicon or to collect more speech?
- Can an objective measure of MCD be converted into a judgment about whether a voice is "good" or "bad"?
- Can this information be used to motivate the user?

To address these questions, our approach is based on exhaustive experimentation of a well studied English database (the ARCTIC slt voice [9]). Non-English languages are evaluated with respect to the range of these results.

3. EXPERIMENTAL RESULTS

The empirical results of this paper fall into five parts. First we investigate the predictive ability of various features sets and conclude that our language-specific features are not required for good performance. Section 3.2 illustrates the effect of database size. Section 3.3 compares the standard phoneme-based voice with a grapheme-based build. Combined, these two experiments provides a simple frame of reference for evaluating other languages, as presented in 3.4. Finally, section 3.5 estimates the relative effectiveness of improving the lexicon versus collecting additional speech.

The computational workload is substantial: the following results have been mined from the creation and evaluation of over five thousand speech synthesizer variants.

3.1. Feature importance in CART tree training

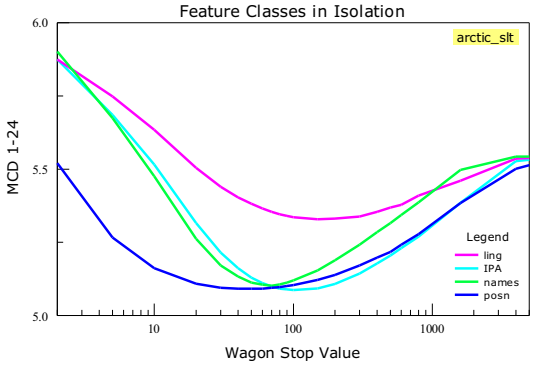

Figure 1. Performance of each feature class in isolation.

Figure 1 plots the mean mel-cepstral distortion curves for the four feature classes trained in isolation. Linguistic information alone is the poorest predictor. Also evident is that the minimum points of the other three curves are all about equal, and lie within one standard deviation (experimentally found to be 0.04). Despite this equality, position features result in a broader basin. For smaller stop values (below 50), IPA and name features are considerably more prone to over-fitting. The relative fragility of feature classes, with position features being most stable predictors, is an aspect that has not been previously noted.

Figure 2 shows the results of combining name features with linguistic, IPA, and position features in pairs. Confirming Figure 1, linguistic features provide no significant reduction in MCD distortion. IPA features do improve the curve somewhat, compared to name features in isolation, but only in the less interesting area of under-training (stop values greater than 80). In contrast, the combination of name and position features results in a substantial improvement of 0.40. Also, the optimal stop value drops from a range of 70-80 to 20-30, indicating that this combination of information is less prone to over-fitting. The extra addition of IPA features results in a barely-significant reduction from 4.74 to 4.70, though it also shifts the optimal point right, and broadens the "valley," as shown in Figure 3.

From these experiments we conclude that the language-neutral name and position features used in combination are sufficient predictors; that IPA features contribute only a modest amount, while linguistic features are not required. Stated another way, in a language with the phonemes /AA/ and /AE/, these could just as well be named Aardvark and Aesop. Knowing that one is a low back unrounded vowel while the other is mid-low front, is not critical.

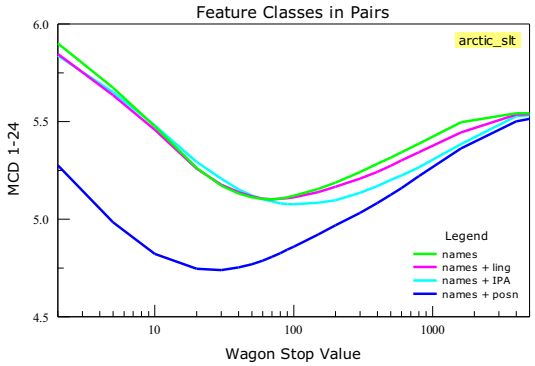

Figure 2. Performance of feature classes combined in pairs, with the single class *names* shown for reference.

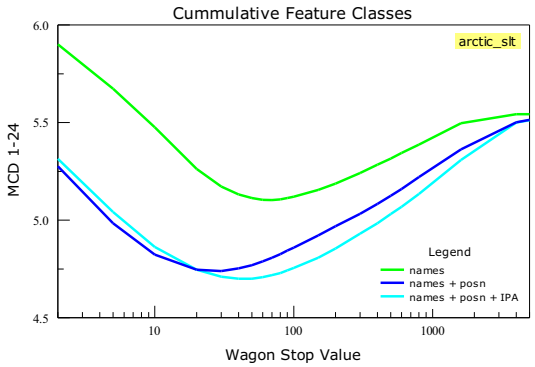

Figure 3. The cumulative effect of incrementally adding name, position, and IPA features. Linguistic features provide no additional improvement.

3.2. Effect of database size

In Section 3.1 all experiments were performed on the same single-speaker database containing slightly less than one hour of speech. The MCD values for this database cannot be directly compared to other voices due to differing amounts of recordings. Venturing the bold assumption that no particular speaker is harder to model than any other, we contend that MCD distortion can be normalized by size. To provide normalization points, the ARCTIC slt database was subdivided into amounts of ½, ¼, 1/8, and 1/16 hour. (As measured by wavefiles lengths. With silence and pauses excluded, the real amount of speech is about 10% less.)

In Figure 4 the effect of database size is seen as a series of layered curves, with a nearly linear decrease in distortion observed as the amount of speech doubles. The voices were trained using name and position features, with data points

calculated as the average of 10-fold experiments of a 90/10% training/heldout split. For sake of comparison, the lowermost, downward-shooting dashed curve is derived by evaluating on the training set. Where it decreases while the other curves turn upward is the region of *over-fitting* (found to the left 20). Notice that the optimal stop value is stable over a wide range of database sizes, a reassuring result.

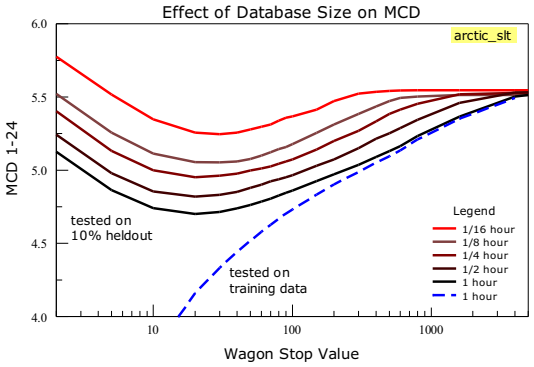

Figure 4. The effect of database size on MCD. For comparison, the lowermost curve tests on the training set.

The series of curves of Figure 4 do not reveal an asymptotic lower limit. Consequently, we do not know when it becomes pointless to collect more speech. Our expectation, based on working with larger English databases, is that MCD distortion levels-out somewhere between 10 and 20 hours. Due to the increased data collection and computational requirements, finding this limit is an effort still in progress.

3.3. Effect of not having a lexicon for English

The results of Section 3.2 provide one half of a calibration reference. The other half is a size-versus-MCD curve from a corresponding *bad* voice. In combination, the two provide lower and upper guide-rails. A voice created in another language can be placed in the context of good-to-bad for a given size, and good-to-bad overall.

To serve this purpose the bad voice cannot be *randomly* bad. The most appropriate thing is to mimic the voice creation experience of a person using the SPICE tools. The equivalent, then, is a version of ARCTIC slt in which the fully detailed lexicon is replaced with very basic letter-to-sound rules. To this end, we've implemented a grapheme-based voice.[3] Given the highly irregular spelling of English, such a voice is trained on a large percentage of mis-pronunciations. Hence, the MCD curve for this voice can be considered a generous upper bound. These two conditions are contrasted in Figure 5, where the vertical distance between them mostly ranges from 0.25 to 0.3. It is reasonable to believe that the vertical gap is a language-dependent attribute. A language with a regular writing system, such as Spanish, will display a narrow gap and, unlike English, not present a useful context for calibrating other languages.

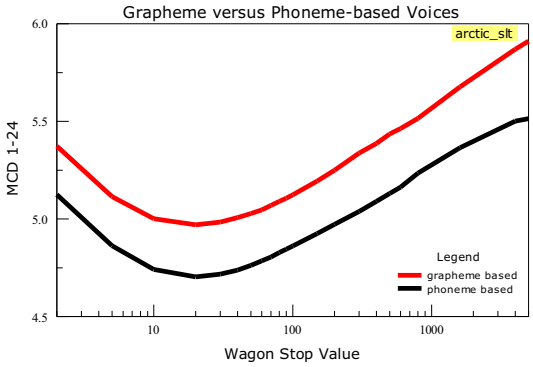

Figure 5. Curves of grapheme-bases versus phoneme-based English voices, built from 1 hour of speech. The difference in MCD at a stop value of 20 is 0.268.

3.4. Evaluating languages other than English

The best form of evaluation is via large-scale listening tests such as is conducted during the Blizzard Challenge events [10]. While it is crucial to have SPICE users listen to their voice during development (e.g. transcription of unseen sentences), it is also valuable to have an automated means of evaluation. Figure 6 places the MCD distortion values of French, German, Tamil, Hindi, Bulgarian, Mandarin, Vietnamese and Konkani voices in context of the upper and lower English curves. Each is trained from 90% of the available data and tested on the residual 10%.

In informal review with the developers, the voices for Vietnamese, Konkani, and Mandarin were deemed to be poor. French, German, and Tamil were deemed good. Hindi and Bulgarian behaved acceptably within-domain but not so well out of domain. In semi-formal listening tests, the Hindi and German voices scored 75% on in-domain word comprehension [4]. These assessments are generally consistent with the picture of Figure 6. That Hindi and Tamil scored so well is notable. While the grapheme-to-phoneme relation of these languages is comparatively straightforward, the low distortion is likely due to their limited domain of application. The 10% heldout sentences contained phrases present in the training data, and so do not thoroughly exercise the voice. Finally, the two tonal languages are inconclusive and are in need of further investigation.

3 During training all numbers are expanded to word form.

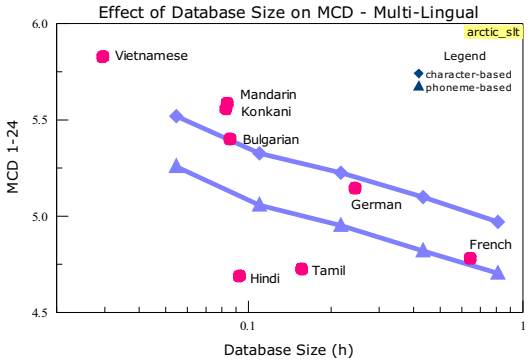

Figure 6. Eight languages placed in context of the bounds of a grapheme-based (upper line) and a phoneme-based English voice (lower line).

3.5. Best use of effort - better lexicon or more speech?

For each of the dots plotted in Figure 6, the pair of English calibration lines allow one to make a rough assessment of voice quality, and to recommend a course of action. The course of action can be: record more data, fix the lexicon, or both. The French voice is already in good shape. The German voice could use an improved lexicon. Hindi and Tamil could benefit from additional speech data. Mandarin, Konkani, and Bulgarian need more data and better lexicons. (Vietnamese, the outlier, had other technical issues.)

Which is more important – improving the lexicon or collecting more speech? For English, the gap between, and slope of the two curves in Figure 6 provides one answer: fixing the lexicon offers an improvement equal to that of collecting five times the amount of recorded speech. In our experience, an efficient and careful voice talent can record three utterances per minute, while the rate of fixing words in the lexicon is maybe twice that. Table 2 suggests that when the database is small (less than half an hour) it is better to record more speech, saving lexical work until later.

approx. data size (h)	*number utter-ances*	*number unique words*	*Δ word/ Δ utt. ratio*	*effect of fixing lexicon*	*double speech data*
1/16	77	384	4.99	0.261	—
1/8	154	645	3.39	0.270	0.2012
1/4	311	1103	2.92	0.273	0.1047
1/2	607	1770	2.25	0.280	0.1319
1	1132	2766	1.92	0.268	0.1176

Table 2. A comparison of the MCD improvement from either fixing the lexicon or doubling the speech database size.

4. CONCLUSION

While expert attention to phoneme definition and lexicon tuning can greatly improve voice quality, for non-expert users extra data collection may be an easier route to success. We are planning listening tests to better correlate MCD with intelligibility and perceived quality scores.

Returning to the questions posed in Section 2.4, our data leads us to conclude that a) our language-dependent training features are unimportant, b) a doubling of speech data results in a drop in MCD of 0.12, c) at a fixed size, the difference between grapheme- and phoneme-based English voices is 0.27, d) collecting additional speech is more time-efficient than lexicon correction, up to about one hour of data, e) MCD can be used to estimate the quality of voices in new languages (as in Figure 6), and f) yes, it is motivating.

However, global MCD measurements do not indicate when a *particular* word is mispronounced. Our next goal is to identify such words and judiciously present to the user the minimal number of entries for lexical correction.

5. ACKNOWLEDGMENTS

This work is in part supported by the US National Science Foundation under grant number 0415021 "SPICE: Speech Processing Interactive Creation and Evaluation Toolkit for new Languages." Any opinions expressed in this material are those of the authors and do not necessarily reflect the views of the NSF.

6. REFERENCES

[1] Black, A., and Lenzo, K., *The FestVox Project: Building Synthetic Voices,* 2000, http://festvox.org/bsv.

[2] Schultz, T., Black, A., Badaskar, S., Hornyak, M., Kominek, J., *SPICE: Web-based Tools for Rapid Language Adaptation in Speech Processing Systems*, Interspeech 2007, Antwerp, Belguim.

[3] SPICE, http://cmuspice.org.

[4] Kominek, J., Schultz, T., Black, A. *Voice Building from Insufficient Data – Classroom Experiences with Web-based Language Development Tools*, ISCA Speech Synthesis Workshop 6, Bonn, German, 2007.

[5] Davel, M., Barnard, E., *Efficient generation of pronunciation dictionaries: human factors during bootstrapping*, Interspeech 2004, Jeju, Korea.

[6] Mashimo, M., Toda, T., Shikano, K., Campbell, N., *Evaluation of Cross-language Voice Conversion based on GMM and STRAIGHT*, Eurospeech 2001, Aalborg, Denmark.

[7] EST, http://www.cstr.ed.ac.uk/projects/speech_tools.

[8] Black, Alan W, *CLUSTERGEN: a Statistical Parametric Synthesizer using Trajectory Modeling*, Interspeech 2006, Pittsburgh, PA, USA.

[9] Kominek, J., Black, A., *CMU Arctic Speech Database, Speech Synthesis Workshop 5, Pittsburgh, USA, 2004.*

[10] Black, A., and Tokuda, K., *Blizzard Challenge – 2005: Evaluating corpus-based speech synthesis on common datasets* Interspeech 2005, Lisbon, Portugal.