LSTM Deep Neural Networks Postfiltering for Improving the Quality of Synthetic Voices

Marvin Coto-Jiménez[1,2(✉)] and John Goddard-Close[2]

[1] University of Costa Rica, San José, Costa Rica
marvin.coto@ucr.ac.cr
[2] Autonomous Metropolitan University, Mexico, DF, Mexico
jgc@xanum.uam.mx

Abstract. Recent developments in speech synthesis have produced systems capable of providing intelligible speech, and researchers now strive to create models that more accurately mimic human voices. One such development is the incorporation of multiple linguistic styles in various languages and accents. HMM-based speech synthesis is of great interest to researchers, due to its ability to produce sophisticated features with a small footprint. Despite such progress, its quality has not yet reached the level of the current predominant unit-selection approaches, that select and concatenate recordings of real speech. Recent efforts have been made in the direction of improving HMM-based systems. In this paper, we present the application of long short-term memory deep neural networks as a postfiltering step in HMM-based speech synthesis. Our motivation stems from a desire to obtain spectral characteristics closer to those of natural speech. The results described in the paper indicate that HMM-voices can be improved using this approach.

Keywords: LSTM · HMM · Speech synthesis · Statistical parametric speech synthesis · Postfiltering · Deep learning

1 Introduction

Text-to-speech synthesis (TTS) is the technique of generating intelligible speech from a given text. Applications of TTS have grown from early systems which aid the visually impaired, to in-car navigation systems, e-book readers, spoken dialog systems, communicative robots, singing speech synthesizers, and speech-to-speech translation systems [1].

More recently, TTS systems have moved from the task of producing intelligible voices, to the more difficult challenge of generating voices in multiple languages, with different styles and emotions [2]. Despite these trends, there are unresolved obstacles, such as improving the overall quality of the voices. Some researchers are striving to create TTS systems which try to mimic natural human voices more closely.

The statistical methods for TTS, which arose in the late 1990s, have grown in popularity [3], particularly those based on Hidden Markov Models (HMMs).

J.F. Martínez-Trinidad et al. (Eds.): MCPR 2016, LNCS 9703, pp. 280–289, 2016.
DOI: 10.1007/978-3-319-39393-3_28

HMMs are known for their flexibility in changing speaker characteristics, having a low footprint, and their capacity to produce average voices. Previously, HMMs were utilized extensively in the inverse task to TTS of speech recognition. Here they have proved to be successful at providing a robust representation of the main events into which speech can be segmented [4], using efficient parameter estimation algorithms.

More than twenty statistical speech synthesis implementations have been developed for several different languages from around the world. For example [5–16], are a few of the recent publications. Every implementation of a new language, or one of it's dialects, requires the adaptation of HMM-related algorithms by incorporating their own linguistic specifications, and making a series of decisions regarding the type of HMM, decision trees, and training conditions.

In this paper, we present our implementation of a statistical parametric speech synthesis system based on HMM, together with the use of long short-term memory postfilter neural networks for improving its spectral quality.

The rest of this paper is organized as follows: Sect. 2 provides some details of an HMM-based speech synthesis system and in Sect. 3, long short-term memory neural networks are briefly described. Section 4 gives the proposed system and the experiments carried out in order to test the postfilter. Section 5 presents and discusses the results and objective evaluations conducted, and finally, some conclusions are given in Sect. 6.

2 Speech Synthesis Based on HMM

An HMM is a Markov process with unobserved or hidden states. The states themselves emit observations according to certain probability distributions.

In Fig. 1, a representation of a left-to-right HMM is shown, where there is a first state to the left from which transitions can occur to the same state or to the next one on the right, but not in the reverse direction. In this p_{ij} represents the probability of transition from state i to state j, and O_k represents the observation emitted in state k.

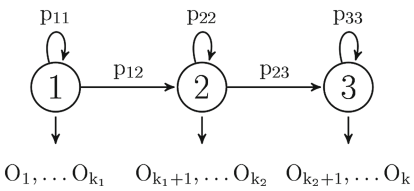


Fig. 1. Left to right example of an HMM with three states

In HMM-based speech synthesis, the speech waveforms can be reasonably reconstructed from a sequence of acoustic parameters learnt and emitted as vectors from the HMM states [1]. Typical implementation of this model includes vectors of observations comprising of the pitch, $f0$, the mel frequency cepstral

coefficients, MFCC and their delta and delta features, for an adequate modeling of the dynamic features of speech. A common tool used to build these HMM-based speech systems is known as HTS [17], which we also use in this paper.

In order to improve the quality of the results, some researchers have recently experimented with postfiltering stages, in which the parameters obtained from HTS voices have been enhanced using deep generative architectures [18–21], for example restricted boltzmann machines, deep belief networks, bidirectional associative memories, and recurrent neural networks (RNN).

In the next section, we present our proposal to incorporate long short-term memory recurrent neural networks in order to improve the quality of HMM-based speech synthesis.

3 Long Short-Term Memory Recurrent Neural Networks

Among the many new algorithms developed to improve some tasks related to speech, such as speech recognition, several groups of researchers have experimented with the use of Deep Neural Networks (DNN), giving encouraging results. Deep learning, based on several kinds of neural networks with many hidden layers, have achieved interesting results in many machine learning and pattern recognition problems. The disadvantage of using such networks is they cannot directly model the dependent nature of each sequence of parameters with the former, something which is desirable in order to imitate human speech production. It has been suggested that one way to solve this problem is to include RNN [22,23] in which there is feedback from some of the neurons in the network, backwards or to themselves, forming a kind of memory that retains information about previous states.

An extended kind of RNN, which can store information over long or short time intervals, has been presented in [24], and is called long short-term memory (LSTM). LSTM was recently successfully used in speech recognition, giving the lowest recorded error rates on the TIMIT database [25], as well as in other applications to speech recognition [26]. The storage and use of long-term and short-term information is potentially significant for many applications, including speech processing, non-Markovian control, and music composition [24].

In a RNN, output vector sequences $\mathbf{y} = (y_1, y_2, \ldots, y_T)$ are computed from input vector sequences $\mathbf{x} = (x_1, x_2, \ldots, x_T)$ and hidden vector sequences $\mathbf{h} = (h_1, h_2, \ldots, h_T)$ iterating Eqs. 1 and 2 from 1 to T [22]:

$$h_t = \mathcal{H}\left(\mathbf{W}_{xh}x_t + \mathbf{W}_{hh}h_{t-1} + b_h\right) \tag{1}$$

$$y_t = \mathbf{W}_{hy}h_t + b_y \tag{2}$$

where $\mathbf{W}_{ij}$ is the weight matrix between layer i and j, b_k is the bias vector for layer k and $\mathcal{H}$ is the activation function for hidden nodes, usually a sigmoid function $f : \mathbb{R} \rightarrow \mathbb{R}$, $f(t) = \frac{1}{1+e^{-t}}$.

Each cell in the hidden layers of a LSTM, has some extra gates to store values: an input gate, forget gate, output gate and cell activation, so values can be stored in the long or short term. These gates are implemented following the equations:

$$i_t = \sigma\left(\mathbf{W}_{xi}x_t + \mathbf{W}_{hi}h_{t-1} + \mathbf{W}_{ci}c_{t-1} + b_i\right) \tag{3}$$

$$f_t = \sigma\left(\mathbf{W}_{xf}x_t + \mathbf{W}_{hf}h_{t-1} + \mathbf{W}_{cf}c_{t-1} + b_f\right) \tag{4}$$

$$c_t = f_t c_{t-1} + i_t \tanh\left(\mathbf{W}_{xc}x_t + \mathbf{W}_{hc}h_{t-1} + b_c\right) \tag{5}$$

$$o_t = \sigma\left(\mathbf{W}_{xo}x_t + \mathbf{W}_{ho}h_{t-1} + \mathbf{W}_{co}c_t + b_o\right) \tag{6}$$

$$h_t = i_t \tanh\left(c_t\right) \tag{7}$$

where σ is the sigmoid function, i is the input gate activation vector, f the forget gate activation function, o is the output gate activation function, and c the cell activation function. $\mathbf{W}_{mn}$ are the weight matrices from each cell to gate vector.

4 Description of the System

Often, the resulting voices from the HTS system have notable differences with the original voices used in their creation. It is possible to reduce the gap between natural and artificial voices by additional learning directly applied to the data [18]. In our proposal, we use aligned utterances from natural and synthetic voices produced by the HTS system to establish a correspondence between each frame.

Given a sentence spoken using natural speech and also with the voice produced by the HTS, we extract a representation consisting of one coefficient for f0, one coefficient for energy, and 39 MFCC coefficients, using the system Ahocoder [27]. The inputs to the LSTM network correspond to the MFCC parameters of each frame for the sentences spoken using the HTS voice, while the output corresponds to the MFCC parameters given by the natural voice for the same sentence. In this way, we have an exact correspondence given by the alignment between the vectors from each utterance using the HTS voice and the natural voice.

Hence, each LSTM network attempts to solve the regression problem of transforming the values of the speech produced by the artificial and natural voices. This allows a further improvement to the quality of newly synthesized utterances with HTS, and uses the network as a way of refining these synthetic parameters to more closely resemble those of a natural voice. Figure 2 outlines the proposed system.

4.1 Corpus Description

The CMU_Arctic databases were constructed at the Language Technologies Institute at Carnegie Mellon University. They are phonetically balanced, with several US English speakers. It was designed for unit selection speech synthesis research.

The databases consist of around 1150 utterances selected from out-of-copyright texts from Project Gutenberg. The databases include US English male and female speakers. A detailed report on the structure and content of the database and the recording conditions is available in the Language Technologies Institute Tech Report CMU-LTI-03-177 [28]. Four of the available voices were selected: BDL (male), CLB (female), RMS (male) and SLT (female).

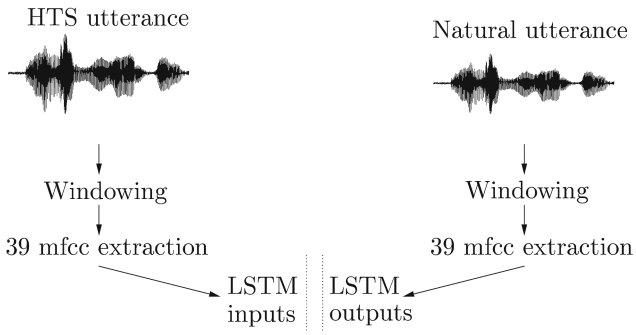


Fig. 2. Proposed system. HTS and Natural utterances are aligned frame by frame

4.2 Experiments

Each voice was parameterized, and the resulting set of vectors was divided into training, validation, and testing sets. The amount of data available for each voice is shown in Table 1. Despite all voices uttering the same phrases, the length differences are due to variations in the speech rate of each speaker.

Table 1. Amount of data (vectors) available for each voice in the databases

Database	Total	Train	Validation	Test
BDL	676554	473588	135311	67655
SLT	677970	474579	135594	67797
CLB	769161	538413	153832	76916
RMS	793067	555147	158613	79307

The LSTM networks for each voice had three hidden layers, with 200, 160 and 200 units in each one respectively.

To determine the improvement in the quality of the synthetic voices, several objective measures were used. These measures have been applied in recent speech synthesis experiments and were found to be reliable in measuring the quality of synthesized voices [29,30]:

– Mel Cepstral Distortion (MCD): Excluding silent phonemes, between two waveforms v^{targ} and v^{ref} it can be measured following Eq. 8 [31]

$$\text{MCD}\left(v^{\text{targ}}, v^{\text{ref}}\right) = \frac{\alpha}{T}\sum_{t=0}^{T-1}\sqrt{\sum_{d=s}^{D}\left(v_d^{\text{targ}}(t) - v_d^{\text{ref}}(t)\right)^2} \tag{8}$$

where $\alpha = \frac{10\sqrt{2}}{\ln 10}$, T is the number of frames of each utterance, and D the total number of parameters of each vector.

– mfcc trajectory and spectrogram visualization: Observation of these figures allow a simple visual comparison between the similitude of the synthesized and natural voices.

These measures were applied to the test set after being processed with the LSTM networks, and the results were compared with those of the HTS voices. The results and analysis are shown in the following section.

5 Results and Analysis

For each synthesized voice produced with HTS and processed with LSTM networks, MCD results are shown in Table 2. It can be seen how this parameter improved when all voices were processed with LSTM networks.

This shows the ability of these networks to learn the particular regression problem of each voice.

Table 2. MCD between HTS and natural voices, and between LSTM postfiltering and natural voices

Database	HTS to natural	LSTM-pf to natural
BDL	8.46	7.98
CLB	7.46	6.87
SLT	7.03	6.65
RMS	7.66	7.60

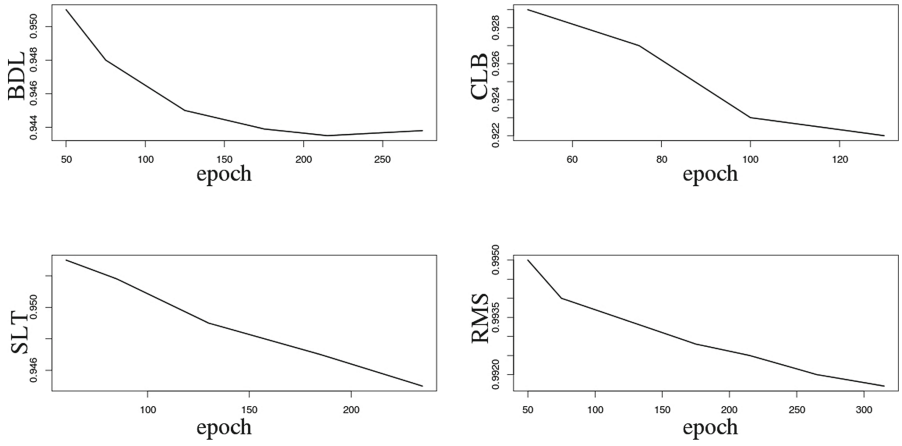


Fig. 3. Evolution of MCD improvement in LSTM postfiltering during training epochs

The best result of MCD improvement with the LSTM postfiltering is CLB (11.2 %) and the least best was RMS (1 %). Figure 3 shows how MCD evolves with the training epochs for each voice. All HTS voices, except one, were improved by the LSTM neural network postfilter for MCD after the first 50 epochs of training.

The differences in the number of epochs required to reach convergence in each case are notable. This can be explained by the difference in MCD between HTS

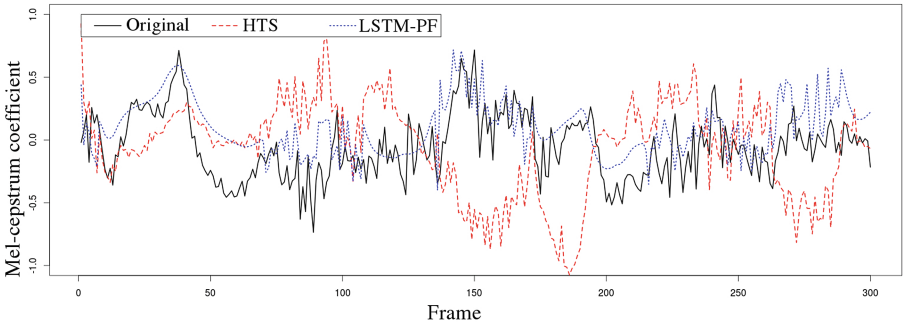
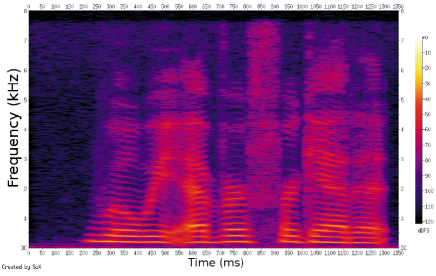
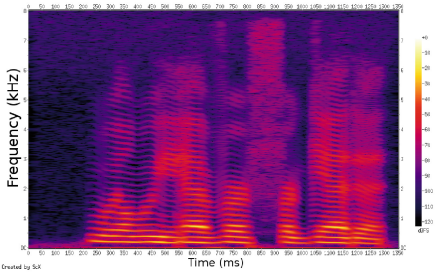


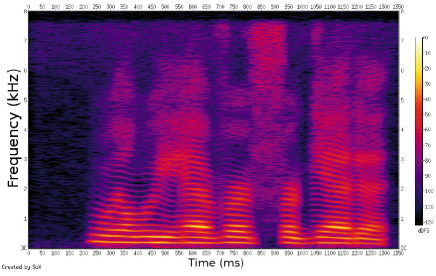
Fig. 4. Illustration of enhancing the 5th mel-cepstral coefficient trajectory by LSTM postfiltering



(a) Original



(b) HTS



(c) LSTM Postfiltering

Fig. 5. Comparison of spectrograms

and natural voices. The gap between them is variable and the LSTM network requires more epochs to model the regression function between them.

An example of the parameters generated by the HTS and the enhancement pursuit by the LSTM postfilter is shown in Fig. 4. It can be seen how the LSTM postfilter fits the trajectory of the mfcc better than the HTS base system.

In Fig. 5 a comparison of three spectrograms of the utterance "Will we ever forget it?" for the voices of: (a) Original (b) HTS and (c) LSTM postfilter enhanced, is shown. The HTS spectrogram usually shows bands in higher frequencies not present in the natural voice, and the LSTM postfilter helps to smooth it, making it closer to the spectrogram of the original voice.

6 Conclusions

We have presented a new proposal to improve the quality of synthetic voices based on HMM with LSTM networks. The method shows how to improve an artificial voice and make it mimic more closely the original natural voice in terms of its spectral characteristics.

We evaluated the proposed LSTM postfilter using four voices, two masculine and two feminine, and the results show that all of them were improved for spectral features, such as MCD measurement, spectrograms, and mfcc trajectory generation.

The improvement of the HTS voices in MCD to the original voices were observed from the first training epochs of the LSTM neural network, but the convergence to a minimum distance took many more epochs. Due to the extensive amount of time required to train each epoch, further exploration should determine new network configurations or training conditions to reduce training time.

Future work will include the exploration of new representations of speech signals, hybrid neural networks, and fundamental frequency enhancement with LSTM postfilters.

Acknowledgements. This work was supported by the SEP and CONACyT under the Program SEP-CONACyT, CB-2012-01, No.182432, in Mexico, as well as the University of Costa Rica in Costa Rica.

References

1. Tokuda, K., Nankaku, Y., Toda, T., Zen, H., Yamagishi, J., Oura, K.: Speech synthesis based on hidden markov models. Proc. IEEE **101**(5), 1234–1252 (2013)
2. Black, A.W.: Unit selection and emotional speech. In: Interspeech (2003)
3. Yoshimura, T., Tokuda, T., Masuko, T., Kobayashi, T., Kitamura, T.: Simultaneous modeling of spectrum, pitch and duration in HMM-based speech synthesis. In: Proceedings of the Eurospeech, pp. 2347–2350 (1999)
4. Falaschi, A., Giustiniani, M., Verola, M.: A hidden markov model approach to speech synthesis. In: Proceedings of the Eurospeech, pp. 2187–2190 (1989)

5. Karabetsos, S., Tsiakoulis, P., Chalamandaris, A., Raptis, S.: HMM-based speech synthesis for the greek language. In: Sojka, P., Horák, A., Kopeček, I., Pala, K. (eds.) TSD 2008. LNCS (LNAI), vol. 5246, pp. 349–356. Springer, Heidelberg (2008)
6. Pucher, M., Schabus, D., Yamagishi, Y., Neubarth, F., Strom, V.: Modeling and interpolation of austrian german and viennese dialect in HMM-based speech synthesis. Speech Commun. **52**(2), 164–179 (2010)
7. Erro, D., Sainz, I., Luengo, I., Odriozola, I., Sánchez, J., Saratxaga, I., Navas, E., Hernáez, I.: HMM-based speech synthesis in basque language using HTS. In: Proceedings of the FALA (2010)
8. Stan, A., Yamagishi, Y., King, S., Aylett, M.: The romanian speech synthesis (RSS) corpus: building a high quality HMM-based speech synthesis system using a high sampling rate. Speech Commun. **53**(3), 442–450 (2011)
9. Kuczmarski, T.: HMM-based speech synthesis applied to polish. Speech Lang. Technol. **12**, 13 (2010)
10. Hanzlíček, Z.: Czech HMM-based speech synthesis. In: Sojka, P., Horák, A., Kopeček, I., Pala, K. (eds.) TSD 2010. LNCS, vol. 6231, pp. 291–298. Springer, Heidelberg (2010)
11. Li, Y., Pan, S., Tao, J.: HMM-based speech synthesis with a flexible mandarin stress adaptation model. In: Proceedings of the 10th ICSP2010 Proceedings, Beijing, pp. 625–628 (2010)
12. Phan, S.T., Vu, T.T., Duong, C.T., Luong, M.C.: A study in vietnamese statistical parametric speech synthesis based on HMM. Int. J. **2**(1), 1–6 (2013)
13. Boothalingam, R., Sherlin, S.V., Gladston, A.R., Christina, S.L., Vijayalakshmi, P., Thangavelu, N., Murthy, H.A.: Development and evaluation of unit selection and HMM-based speech synthesis systems for Tamil. In: National Conference on Communications (NCC), pp. 1–5. IEEE (2013)
14. Khalil, K.M., Adnan, C.: Implementation of speech synthesis based on HMM using PADAS database. In: 12th International Multi-Conference on Systems, Signals & Devices (SSD), pp. 1–6. IEEE (2015)
15. Nakamura, K., Oura, K., Nankaku, Y., Tokuda, K.: HMM-based singing voice synthesis and its application to japanese and english. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 265–269 (2014)
16. Roekhaut, S., Brognaux, S., Beaufort, R., Dutoit, T.: Elite-HTS: a NLP tool for French HMM-based speech synthesis. In: Interspeech, pp. 2136–2137 (2014)
17. HMM-based Speech Synthesis System (HTS). http://hts.sp.nitech.ac.jp/
18. Chen, L.H., Raitio, T., Valentini-Botinhao, C., Ling, Z.H., Yamagishi, J.: A deep generative architecture for postfiltering in statistical parametric speech synthesis. IEEE/ACM Trans. Audio, Speech Lang. Process. (TASLP) **23**(11), 2003–2014 (2015)
19. Takamichi, S., Toda, T., Neubig, G., Sakti, S., Nakamura, S.: A postfilter to modify the modulation spectrum in HMM-based speech synthesis. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 290–294 (2014)
20. Takamichi, S., Toda, T., Black, A.W., Nakamura, S.: Modified post-filter to recover modulation spectrum for HMM-based speech synthesis. In: IEEE Global Conference on Signal and Information Processing (GlobalSIP), pp. 547–551 (2014)
21. Prasanna, K.M., Black, A.W.: Recurrent Neural Network Postfilters for Statistical Parametric Speech Synthesis. arXiv preprint (2016). arXiv:1601.07215
22. Fan, Y., Qian, Y., Xie, F.L., Soong, F.K.: TTS synthesis with bidirectional LSTM based recurrent neural networks. In: Interspeech, pp. 1964–1968 (2014)

23. Zen, H., Sak, H.: Unidirectional long short-term memory recurrent neural network with recurrent output layer for low-latency speech synthesis. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 4470–4474 (2015)
24. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. **9**(8), 1735–1780 (1997)
25. Graves, A., Jaitly, N., Mohamed, A.: Hybrid speech recognition with deep bidirectional LSTM. In: IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU) (2013)
26. Graves, A., Fernández, S., Schmidhuber, J.: Bidirectional LSTM networks for improved phoneme classification and recognition. In: Duch, W., Kacprzyk, J., Oja, E., Zadrożny, S. (eds.) ICANN 2005. LNCS, vol. 3697, pp. 799–804. Springer, Heidelberg (2005)
27. Erro, D., Sainz, I., Navas, E., Hernaez, I.: Improved HNM-based vocoder for statistical synthesizers. In: InterSpeech, pp. 1809–1812 (2011)
28. Kominek, J., Black, A.W.: The CMU Arctic speech databases. In: Fifth ISCA Workshop on Speech Synthesis (2004)
29. Zen, H., Senior, A., Schuster, M.: Statistical parametric speech synthesis using deep neural networks. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (2013)
30. Zen, H., Senior, A.: Deep mixture density networks for acoustic modeling in statistical parametric speech synthesis. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (2014)
31. Kominek, J., Schultz, T., Black, A.W.: Synthesizer voice quality of new languages calibrated with mean mel cepstral distortion. In: SLTU (2008)