Deep neural network context embeddings for model selection in rich-context HMM synthesis

Thomas Merritt[1], *Junichi Yamagishi*[1,2], *Zhizheng Wu*[1], *Oliver Watts*[1], *Simon King*[1]

[1]The Centre for Speech Technology Research, University of Edinburgh, United Kingdom
[2]National Institute of Informatics, Japan

t.merritt@ed.ac.uk

Abstract

This paper introduces a novel form of parametric synthesis that uses context embeddings produced by the bottleneck layer of a deep neural network to guide the selection of models in a rich-context HMM-based synthesiser. Rich-context synthesis – in which Gaussian distributions estimated from single linguistic contexts seen in the training data are used for synthesis, rather than more conventional decision tree-tied models – was originally proposed to address over-smoothing due to averaging across contexts. Our previous investigations have confirmed experimentally that averaging across different contexts is indeed one of the largest factors contributing to the limited quality of statistical parametric speech synthesis. However, a possible weakness of the rich context approach as previously formulated is that a conventional tied model is still used to guide selection of Gaussians at synthesis time. Our proposed approach replaces this with context embeddings derived from a neural network.

Index Terms: speech synthesis, hidden Markov model, deep neural networks, rich context, embedding

1. Introduction

HMM speech synthesis systems offer a flexible and adaptable way to synthesise speech. However the naturalness of these systems is consistently rated below natural speech and unit selection systems as observed in the evaluation results from numerous Blizzard Challenges over many years [1, 2, 3, 4]. Many explanations have been given for the causes of this [5, 6, 7, 8] however few formal investigations have been performed.

In previous work, we did formally investigate several hypotheses, including the effects of: over-smoothing of the spectral envelope as a result of averaging over multiple speech samples from differing contexts [9], temporally over-smoothed parameter trajectories as a result of maximum-likelihood parameter generation (MLPG) [10, 11, 9], parameter generation with poor global variance [10, 11, 9], vocoding[12, 11, 9] and independant modelling of parameter streams[12, 11], among others.

The most striking finding of these investigations was that temporal over-smoothing does not have as strong effect on speech quality as was previously believed, but rather that the gap between standard HMM speech synthesis systems and vocoded speech might be significantly closed by avoiding the averaging of speech samples across differing linguistic contexts. Motivated by this finding, we now propose a novel form of statistical parametric speech synthesis that avoids this harmful across-contexts averaging.

2. Motivation

Our previous work has focused on the perceptual effects introduced by modelling in statistical parametric speech synthesis [10, 12, 11, 9]. From these investigations, the effect of averaging across differing linguistic contexts (as is done in all decision tree-clustered HMM synthesis systems) was identified as introducing a very substantial drop in quality. Specifically, in [9] we reported a large perceptual degradation when moving from an oracle condition where mean parameter values were calculated by only averaging within the same context compared to a standard decision tree-clustered tied-parameter HMM system (built as per the HTS demo recipe).

This oracle system calculated the Gaussian mean values (for static, delta and delta-delta acoustic features) from the acoustic features (i.e., vocoder parameters) of a recording of the test sentence. The synthetic speech was then created by using MLPG on this sequence of Gaussian means, with variance values coming from the conventional HMMs. Of course, this oracle system is of no use for actual text-to-speech because it is impossible to have examples in the training data that exactly match all required contexts for every test sentence. However, it motivates the system presented here that – like existing so-called "rich-context" systems – avoids averaging across linguistic contexts to train the Gaussian means.

3. Prior work

There are two notable examples of systems that also aim to remove the effects of across-context averaging. The first is simply unit selection synthesis, where individual tokens are used, without averaging. Unit selection synthesis also avoid vocoding, another limitation on the quality of parametric speech synthesis [12, 11, 9]. The other example is rich-context statistical parametric speech synthesis systems [13, 14, 15, 16].

The term 'rich-context' refers to models which are trained only on samples where the context matches exactly and therefore avoids averaging across differing contexts. The primary example is [13], in which Gaussian mean values are calculated within each unique context found in the training data, with variance values being tied in the usual way[1]. This system would appear to be very close to our previously investigated oracle within-context-averaging system [9]. We now examine how this system selects a suitable rich-context model to use at synthesis time, given that exact matches to the required contexts within a test sentence are extremely unlikely to be available.

[1]In practice, such a system is easy to derive from a conventional tied system, simply by untying all parameters, then performing further training in which only the means are updated.

4. Conventional rich context system

The system introduced in [13] uses the distribution (i.e., Gaussian) selected by the standard tied decision tree as a reference. It then finds the closest untied rich-context model (from a pre-selected subset of all possible models) to this reference, using equation 2 to compute divergence between the reference distribution and each of the rich context models. This equation, as described in [17], is an adapted version of Kullback-Leibler divergence (KLD) for calculating divergence between multi-space probability distribution HMMs (MSD-HMMs) and can therefore be applied to both spectrum (S) and pitch ($f0$) parameters independently.

$$D_{KL}^{S+f0}(p||q) = D_{KL}^{S}(p||q) + D_{KL}^{f0}(p||q), \tag{1}$$

where

$$D_{KL}(p||q) \leq (w_0^p - w_0^q)\log\frac{w_0^p}{w_0^q} + (w_1^p - w_1^q)\log\frac{w_1^p}{w_1^q}$$
$$+\frac{1}{2}tr\{(w_1^p\Sigma_p^{-1} + w_1^q\Sigma_q^{-1})(\mu_p - \mu_q)(\mu_p - \mu_q)^\top$$
$$+w_1^p(\Sigma_p\Sigma_q^{-1} - I) + w_1^q(\Sigma_q\Sigma_p^{-1} - I)\}$$
$$+\frac{1}{2}(w_1^q - w_1^p)\log \mid \Sigma_p\Sigma_q^{-1} \mid, \tag{2}$$

p and q are the reference and pre-selected HMM states respectively, w_0 and w_1 are the prior probabilities of unvoiced and voiced respectively (for spectrum $w_0 \equiv 0$ and $w_1 \equiv 1$), μ and Σ are the mean and covariance of the Gaussian distributions respectively and $|\cdot|$ indicates the determinant of a matrix. The divergence for spectrum ($D_{KL}^{S}(p||q)$) and pitch parameters ($D_{KL}^{f0}(p||q)$) are then summed together using equation 1 to provide the final divergence score $D_{KL}^{S+f0}(p||q)$. These equations are applied in a state-wise fashion. All divergence values across the test phoneme (5 states) are added together to arrive at a single value per phoneme. The rich-context model (i.e., all 5 states in that model come from the same context) with lowest total divergence is then selected

4.1. Implementation issues

One issue encountered when using this formula with rich context models, but unreported in [13], is that rich context models are generally either completely voiced or unvoiced, making either w_0 or w_1 equal to zero. In our replication of Yan2009, where this occurs, a small number (0.001) was added to or subtracted from w_0 and w_1 to ensure a division by zero never takes place. The problem of zero divisions also appears in the spectrum calculation where $w_0 \equiv 0$ and $w_1 \equiv 1$; in this case $(w_0^p - w_0^q)log\frac{w_0^p}{w_0^q}$ was set to 0.

Also the adaptation to enable the KLD algorithm to be used for MSD-HMMs means that the divergence measure is no longer symmetric; so $D_{KL}(p||q)$ and $D_{KL}(q||p)$ were averaged together to give the final divergence score. This was not mentioned in [13], so it can only be assumed that this was how the original implementation was done.

4.2. Critique

The reference distribution used in [13] is a standard tied model. That is, the system chooses the rich context model that is most similar to the model that would be used in a conventional system. This is counter-intuitive. As we know from [9], this tied model is known to be of poor quality as a result of averaging across different contexts and therefore would seem to be a poor reference for rich-context model selection. The whole point of using rich context models is to get away from the tied model, not to find a model that is as close as possible to it.

As mentioned above, the system in [13] selected only from a subset of all possible rich-context models: only contexts matching the triphone of the target context, and if no matches are available this is expanded to biphone match. The need for pre-selection was given as leading to a 'reasonable size of the search space' [13].

5. Proposed bottleneck-driven system

Our proposed system is inspired by that in [13], however does not use the tied model as a reference for rich context model selection. Instead, it performs selection using an acoustically-supervised embedding of the linguistic context, which we derive from the bottleneck layer of a Deep Neural Network (DNN) speech synthesis model [18].

The activations at the bottleneck layer of this network comprise a very compact (e.g., 32-dimensional) feature vector that has been learnt over the training data; such a feature vector is often termed an 'embedding' [19].

Each unique input to the DNN (i.e., each unique linguistic context) leads to a particular bottleneck feature vector. That is, we can derive a compact vector-space representation of any linguistic context, including those not seen in the training data. We use distance in this vector space as the way to select rich-context models at synthesis time. The DNN-derived embedding is essentially a compression of the linguistic features, but importantly one that has been learned in conjunction with predicting the acoustics. So, for example, acoustically-irrelevant linguistic features will be ignored, and other features will be 'de-noised' and de-correlated.

Using the speech parameter space to calculate perceptual distance – which is effectively what the system in [13] does – has been previously raised by Taylor, who is 'uneasy about the use of cepstral space to represent the perceptual space' [20]. In our proposed system, an embedding of the linguistic space, as learnt by a DNN, is used instead.

Our proposed bottleneck-guided rich-context system has the added benefit that we are no longer constrained by needing to use speech parameters at the output layer of the DNN, since it is to be used only for deriving the bottleneck features. For example, we could use perceptually-motivated features instead of vocoder parameters; this is future work.

Various measures could be used, at synthesis time, to find the closest rich-context model (in bottleneck feature space) for an unseen context, this can be done in a number of ways. Here we present two possibilities: Euclidean distance and KLD. First, we give more details of how the bottleneck features are derived.

5.1. Bottleneck features

To generate bottleneck features, we used a feed-forward neural network with six hidden layers. Each layer had 1024 hidden units except that the second hidden layer was set as a bottleneck layer which had only 32 hidden units, as in our preliminary experiments we found that using the second layer as the bottleneck layer achieved the best performance for DNN-based speech synthesis in terms of acoustic feature distortions. More details about the input and output features and implementations of DNN can be found in [18]. The input to the DNN

includes HMM state-position (i.e., sub-phoneme) and frame-within-current-state counter-based features. Bottleneck features were pre-computed for all frames in the training data using a forward pass, and the mean and variance of the features was computed per rich-context HMM state; these distributions were then stored with the rich-context HMM.

5.2. Euclidean distance selection

The Euclidean distance between the bottleneck features computed for a given state in the test sentence, and the stored features for all the rich-context states is:

$$D(b,g) = \sum_{n=1}^{N} || \; b_n - \mu_g \; ||_2, \tag{3}$$

where b is the frame-level sequence of bottleneck features for the current state in the test sentence, g is the Gaussian distribution (with mean μ_g) of bottleneck features for a candidate rich-context model and N is the duration (in frames) of the current test sentence HMM state. The distance for each state in the current phoneme is summed and the phone-sized rich-context model with the smallest Euclidean distance is selected (i.e., all 5 states are taken from the same rich-context model).

5.3. Kullback Leibler divergence selection

The KLD [21] between distribution f of the bottleneck features computed for the frames corresponding to a given state in the test sentence, and distribution g, is calculated as:

$$D_{KL}(f||g) = \frac{1}{2}[log\frac{\mid \Sigma_g \mid}{\mid \Sigma_f \mid} + Tr[\Sigma_g^{-1}\Sigma_f] - d$$

$$+(\mu_f - \mu_g)^T \Sigma_g^{-1}(\mu_f - \mu_g)], \tag{4}$$

where μ and Σ are mean and covariance and d is the dimensionality (32 in this case) of the bottleneck feature vector. As with the Euclidean distance, the KLD for each sub-phonetic state is summed over the phoneme and the closest model chosen. Variance values were floored to 1% of the global variance. A symmetric version of KLD was used in practice: the average of $D_{KL}(f||g)$ and $D_{KL}(g||f)$.

6. Experiments

6.1. Implementation

We built a variety of system configurations, shown in Table 1, and compared them in a listening test. We created a best-effort replication of the system described in [13], one with tri-phone (backing off to bi-phone where necessary) pre-selection as per the original system and another with more relaxed bi-phone (backing off to mono-phone where necessary) pre-selection. The latter system has a wider set of rich-context models to select from, per test sentence phoneme, and so should be able to choose a model that is closer to the tied model reference. For that reason, we hypothesise that this will actually sound worse than the more constrained system (even though the reasons for pre-selection given in [13] were only in regard of computational cost). The average pre-selection candidate set size over a set of test sentences is shown in Table 2.

No pre-selection constraints were used in any of the proposed systems (E, KL, ETS, KLTS), to fully test the ability of the DNN to 'embed' the required linguistic information into the bottleneck features.

Table 1: *Conditions included in listening test*

ID	Description	Postfilter
N	Natural speech	n/a
V	Vocoded speech	n/a
D	Stacked bottleneck DNN system [18]	PF
H	Standard tied HMM speech (HTS demo)	GV
F	HMM speech w/ fully untied tree (MDL = 0) – variance parameters from system H	PF
CT	Rich context system [13] – tri-phone pre-selection	PF
CB	Rich context system [13] – bi-phone pre-selection	PF
E	Proposed system w/ Euclidean distance (section 5.2)	PF
KL	Proposed system w/ KLD (section 5.3)	PF
ETS	Proposed system w/ Euclidean distance (section 5.2) – source parameters from system H	PF
KLTS	Proposed system w/ KLD (section 5.3) – source parameters from system H	PF

For comparison, a rich-context system guided by a decision tree (rather than the method in [13] or our proposed method) was created (system F) by growing the decision tree with the MDL factor set to 0; this tree has one leaf per unique context seen in the training data. Variances were borrowed from the standard tied system (H).

2400 sentences from a male speaker of British English were used for training all systems [22]. 60 unseen Harvard sentences were used for testing. STRAIGHT [23] was used for speech analysis and the postfilter scaling factor was fixed to 1.2 for all systems (where applied). For all systems, natural durations derived by forced alignment, were used. Before presentation to listeners, all utterances were volume normalised [24]. The decision of whether to use a postfilter or GV was made case-by-case for each system, choosing whichever sounded best in informal listening.

6.2. Experimental setup

The listening test was conducted using the MUSHRA methodology [25], with the same set up as [12]. In MUSHRA, the same sentence is presented to the listener under all conditions, on a single screen. Each condition is then scored between 0 (completely unnatural) to 100 (completely natural). This paradigm was originally developed to evaluate audio codecs and therefore allows small differences between conditions to be heard by the listener and for them to use knowledge of the full range of conditions when rating each one. Natural speech is provided as a hidden (i.e., listeners are not told which condition this is) reference and acts as an upper anchor. Listeners are instructed to rate natural speech, which appears in a random position among the conditions, at 100. Ordinarily there is also a lower anchor when this paradigm is used for audio codecs; however a lower anchor is to difficult to define for synthetic speech so was not used here (as was also the case in [12]).

Stimuli played to listeners along with listener responses can be found at [26].

Table 2: *Average candidates per state over a test set*

	CT	CB
overall average	35	196
tri-phone search	29	n/a
bi-phone search	54	193
centre phone search	982	982

7. Results

Results from the MUSHRA test can be seen in Figures 1 and 2 showing the absolute values awarded to the conditions and rank order respectively. The dashed green lines added to the box plots show the mean values. All tests for significant differences between conditions applied Holm-Bonferroni correction due to the large number of condition pairs to compare. All conditions are significantly different from all others in absolute rating, except between: H and F, KL and E, KL and ETS, KL and KLTS, ETS and KLTS. Significant differences are in agreement using a t-test or Wilcoxon signed-rank test at a p value of 0.05. There is a disagreement in statistical significance between conditions F and E: the Wilcoxon signed-rank test finds the difference in judgements to be statistically significant whereas the t-test doesn't. All conditions are significantly different in rank order except between: H and F, KL and E, ETS and KLTS. These significant differences are in agreement using the Mann-Whitney U test and the Wilcoxon signed-rank test at a p value of 0.05. There is a disagreement in statistical significance between conditions H and E: the Mann-Whitney U test finds the difference in judgements to be statistically significant whereas the Wilcoxon signed-rank test doesn't.

One point of surprise is the ratings given to the CT condition. In informal listening, we found this to be of higher quality than the H and F conditions; it is possible that our expert listener judgement is out of line with the paid listeners' non-expert opinions on naturalness [27]. We suspect that the CT system removes much of the buzzy quality present in system H, but in doing so has made other imperfections audible which are otherwise masked by this buzziness, therefore reducing the perceptual scores from naïve listeners.

No significant improvement in absolute score is observed when source parameters (log fundamental frequency and band aperiodicity) from the standard tied models are used in systems ETS and KLTS compared with KL. This indicates that by performing a KLD search for suitable rich context models we are already incorporating some prosodic information (close to that of tied models). The wide range of scores shown on the boxplot for systems V to KLTS shows that this task of scoring these systems is difficult, presumably because they are all of quite high naturalness and these variances are caused by different systems being better or worse at differing sentences presented.

The difference in naturalness between systems CT and CB indicate that the pre-selection implemented in [13] also steers the system towards selecting better models. This highlights the shortcomings of the reference tied model (system H) used in this system. Conversely, the proposed methods (conditions E, ETS, KL & KLTS), which perform a global search over the training corpus using bottleneck features which embed linguistic context information, allow these systems to select better rich-context models.

The stacked DNN bottleneck synthesis system presented in [18] outperforms all statistical parametric systems tested in this investigation, indicating that, while great improvements in quality have been made to HMM synthetic speech, more work is required. Finally, as already found in previous investigations [11, 12, 9], Figure 1 shows that vocoded speech is already significantly less natural than the original waveform.

8. Conclusions and future work

The proposed system provides clear improvements on both standard tied HMM models and the previously proposed rich context model system [13]. Although a state-of-the-art DNN setup is better than all HMM systems here, there is further room for improvement in the HMM systems, including the use of an embedding that is specifically designed for the task, not just derived from a DNN that was optimised for synthesis. For example, embeddings could be derived from a DNN that no longer needs to output speech parameters, but perhaps uses more perceptually relevant output features.

The HMM paradigm is much more transparent that the DNN paradigm. Rich-context model parameters can be related directly back to frames in the training data, allowing diagnosis and fault-finding to be carried out. This link to the training data also suggest simple and obvious ways to build hybrid systems (i.e., statistical model-guided concatenation). A final suggestion for future work would be to investigate speaker adaptation for a rich-context HMM-based system.

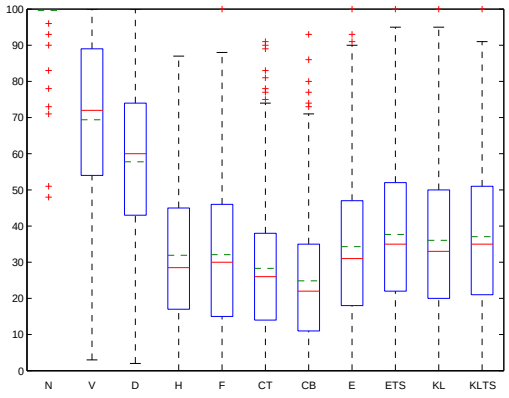

Figure 1: *Boxplot of absolute values given from MUSHRA test*

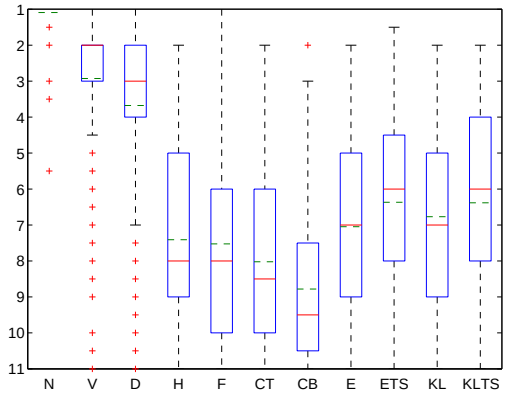

Figure 2: *Boxplot of rank order of conditions from MUSHRA test*

9. Acknowledgements

Thanks to Gustav Eje Henter for help with MUSHRA testing implementation and analysis. This work was supported by EPSRC Programme Grant EP/I031022/1 (Natural Speech Technology).

10. References

[1] S. King and V. Karaiskos, "The Blizzard Challenge 2010," in *Proc. Blizzard Challenge*, 2010.

[2] ——, "The Blizzard Challenge 2011," in *Proc. Blizzard Challenge*, 2011.

[3] ——, "The Blizzard Challenge 2012," in *Proc. Blizzard Challenge*, 2012.

[4] S. King, "Measuring a decade of progress in text-to-speech," *Loquens*, vol. 1, no. 1, 2014.

[5] T. Toda and K. Tokuda, "A Speech Parameter Generation Algorithm Considering Global Variance for HMM-Based Speech Synthesis," *IEICE Transactions on Information and Systems*, vol. E90-D, no. 5, pp. 816–824, May 2007.

[6] S. King, "An introduction to statistical parametric speech synthesis," *Sadhana*, vol. 36, pp. 837–852, 2011.

[7] K. Tokuda, T. Yoshimura, T. Masuko, T. Kobayashi, and T. Kitamura, "Speech parameter generation algorithms for HMM-based speech synthesis," *Proc. ICASSP*, 2000.

[8] H. Zen, K. Tokuda, and A. W. Black, "Statistical parametric speech synthesis," *Speech Communication*, vol. 51, no. 11, pp. 1039–1064, Nov. 2009.

[9] T. Merritt, J. Latorre, and S. King, "Attributing modelling errors in HMM synthesis by stepping gradually from natural to modelled speech," in *Proc. ICASSP*, 2015.

[10] T. Merritt and S. King, "Investigating the shortcomings of HMM synthesis," in *Proc. 8th ISCA Speech Synthesis Workshop*, 2013, pp. 165–170.

[11] T. Merritt, T. Raitio, and S. King, "Investigating source and filter contributions, and their interaction, to statistical parametric speech synthesis," in *Proc. Interspeech*, 2014, pp. 1509–1513.

[12] G. E. Henter, T. Merritt, M. Shannon, C. Mayo, and S. King, "Measuring the perceptual effects of modelling assumptions in speech synthesis using stimuli constructed from repeated natural speech," in *Proc. Interspeech*, 2014, pp. 1504–1508.

[13] Z.-J. Yan, Y. Qian, and F. K. Soong, "Rich context modeling for high quality HMM-based TTS," in *Proc. Interspeech*, 2009, pp. 1755–1758.

[14] ——, "Rich-context unit selection (RUS) approach to high quality TTS," in *Proc. ICASSP*, 2010, pp. 4798–4801.

[15] S. Takamichi, T. Toda, Y. Shiga, H. Kawai, S. Sakti, and S. Nakamura, "An Evaluation of Parameter Generation Methods with Rich Context Models in HMM-Based Speech Synthesis," in *Proc. Interspeech*, 2012, pp. 1139–1142.

[16] S. Takamichi, T. Toda, Y. Shiga, S. Sakti, G. Neubig, S. Nakamura, and S. Member, "Parameter Generation Methods With Rich Context Models for High-Quality and Flexible Text-To-Speech Synthesis," *Selected Topics in Signal Processing, IEEE Journal of*, vol. 8, no. 2, pp. 239–250, 2014.

[17] H. Liang, Y. Qian, F. K. Soong, and G. Liu, "A cross-language state mapping approach to bilingual (Mandarin-English) TTS," in *Proc. ICASSP*, 2008, pp. 4641–4644.

[18] Z. Wu, C. Valentini-Botinhao, O. Watts, and S. King, "Deep neural networks employing multi-task learning and stacked bottleneck features for speech synthesis," in *ICASSP*, 2015.

[19] S. Bengio and G. Heigold, "Word Embeddings for Speech Recognition," in *Proc. Interspeech*, 2014, pp. 1053–1057.

[20] P. Taylor, "The target cost formulation in unit selection speech synthesis." in *Proc. Interspeech*, 2006, pp. 2038–2041.

[21] J. R. Hershey and P. a. Olsen, "Approximating the Kullback-Leibler divergence between Gaussian mixture models," in *Proc. ICASSP*, 2007.

[22] M. Cooke, C. Mayo, and C. Valentini-Botinhao, "Hurricane natural speech corpus, [sound]," LISTA Consortium, doi:10.7488/ds/140, 2013.

[23] H. Kawahara, "STRAIGHT, exploitation of the other aspect of VOCODER: Perceptually isomorphic decomposition of speech sounds," *Acoust. Sci. Technol.*, vol. 27, no. 6, pp. 349–353, 2006.

[24] *Objective measurement of active speech level*, ITU Recommendation ITU-T P.56, International Telecommunication Union, Telecommunication Standardization Sector, Geneva, Switzerland, March 2011.

[25] *Method for the subjective assessment of intermediate quality level of coding systems*, ITU Recommendation ITU-R BS.1534-1, International Telecommunication Union Radiocommunication Assembly, Geneva, Switzerland, March 2003.

[26] T. Merritt, J. Yamagishi, Z. Wu, O. Watts, and S. King, "Listening test materials for "deep neural network context embeddings for model selection in rich-context hmm synthesis", 2015 [dataset]," university of Edinburgh, The Centre for Speech Technology Research (CSTR), doi:10.7488/ds/256.

[27] M. Wester, C. Valentini-Botinhao, and G. E. Henter, "Are we using enough listeners? No! an empirically-supported critique of Interspeech 2014 TTS evaluations," in *Proc. Interspeech*, 2015.