From HMMs to DNNs: Where Do the Improvements Come From?

Citation for published version:
Watts, O, Henter, GE, Merritt, T, Wu, Z & King, S 2016, From HMMs to DNNs: Where Do the Improvements Come From? in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).* Institute of Electrical and Electronics Engineers (IEEE), pp. 5505-5509, 41st IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2016, Shanghai, China, 20/03/16. https://doi.org/10.1109/ICASSP.2016.7472730

Digital Object Identifier (DOI):
10.1109/ICASSP.2016.7472730

Link:
Link to publication record in Edinburgh Research Explorer

Document Version:
Peer reviewed version

Published In:
2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)

FROM HMMS TO DNNS: WHERE DO THE IMPROVEMENTS COME FROM?

Oliver Watts, Gustav Eje Henter, Thomas Merritt, Zhizheng Wu, Simon King

The Centre for Speech Technology Research, University of Edinburgh, United Kingdom

ABSTRACT

Deep neural networks (DNNs) have recently been the focus of much text-to-speech research as a replacement for decision trees and hidden Markov models (HMMs) in statistical parametric synthesis systems. Performance improvements have been reported; however, the configuration of systems evaluated makes it impossible to judge how much of the improvement is due to the new machine learning methods, and how much is due to other novel aspects of the systems. Specifically, whereas the decision trees in HMM-based systems typically operate at the state-level, and separate trees are used to handle separate acoustic streams, most DNN-based systems are trained to make predictions simultaneously for all streams at the level of the acoustic frame. This paper isolates the influence of three factors (machine learning method; state vs. frame predictions; separate vs. combined stream predictions) by building a continuum of systems along which only a single factor is varied at a time. We find that replacing decision trees with DNNs and moving from state-level to frame-level predictions both significantly improve listeners' naturalness ratings of synthetic speech produced by the systems. No improvement is found to result from switching from separate-stream to combined-stream predictions.

Index Terms— speech synthesis, hidden Markov model, decision tree, deep neural network

1. INTRODUCTION

Decision tree clustered hidden Markov models (HMMs) have dominated statistical parametric speech synthesis (SPSS) for the past decade [1]. [2] reports two factors (among others) that limit the quality of HMM-based speech synthesis: 1) state-level averaging within matching linguistic contexts; and 2) state-level averaging across differing linguistic contexts. The first type of averaging inherent to HMMs slightly degrades performance, while the second type of averaging which results from decision tree based clustering is very harmful. However, deep neural networks (DNNs) have recently received attention as a powerful alternative acoustic model for SPSS. A number of studies have demonstrated that DNNs can achieve significantly better performance than decision tree clustered HMMs [3, 4, 5, 6]. However, in moving from HMM- to DNN-based systems, researchers have typically simultaneously modified several aspects of systems besides the machine learning method (decision tree/DNN) used to map from linguistic features to acoustics. For example, decision trees in HMM-based systems make predictions for each hidden state, and these predictions are made separately for source and vocal-tract filter parameters. DNN systems, in contrast, typically operate directly at the level of the acoustic frame, and are typically trained to output source and vocal tract filter parameters simultaneously. These typical configurations for HMM and DNN-based systems are represented by the first and last lines of Table 1, respectively.

Some of these characteristics are a natural accompaniment to the choice of machine learning method. For example, the motivation for training separate decision trees for different streams in HMM-based systems is due to the fact that different linguistic features are expected to be important for different streams. This means that making combined predictions for multiple streams with a single decision tree would be problematic, as questions relevant only to one stream would partition the data in a suboptimal way for the other streams if asked near the root of the tree. However, the fact that moving from a typical HMM system to a typical DNN one means changing several such factors in the design of the system is scientifically unsatisfactory, as it makes it hard to know which factors are responsible for any performance gains. This paper seeks to disentangle performance gains due to three main factors that differ between paradigms, by building a continuum of systems along which only one factor is altered at a time, as shown in Table 1:

Table 1. *Continuum between standard HMM and DNN systems*

System	Regression model	Regression target unit	Stream modelling
1 (HMM)	decision tree	state	separate
2	neural network	state	separate
3	neural network	state	combined
4	neural network	frame	separate
5 (DNN)	neural network	frame	combined

Note that the configurations in Table 1 do not include all possible 8 combinations of the 3 factors. Decision tree systems with combined stream predictions are excluded because we would expect them to perform particularly poorly, as already discussed. Neither was any system built to cover the case of a frame-level decision tree system with separate streams, even though this is an established approach [7]. The combinations considered suffice for us to move one step at a time from a typical HMM-based system towards a typical DNN-based one, and lines 3 and 4 of Table 2 represent two different pathways for making this transition.

2. FROM HMMS TO DNNS

In this section we discuss in greater detail each of the major factors mentioned in Section 1 which will be varied between systems built. We also describe two other more minor factors considered in the experiment.

2.1. Regression model

[3] and [8] outline some shortcomings of decision trees. For example, they are inefficient at expressing complex dependencies between linguistic features, such as the *XOR* relationship, and each model parameter is learned on only a subset of training data. The decision-tree based clustering may also introduce across-linguistic-context averaging, which is found to substantially degrade the naturalness of synthesised speech [2]. Deep neural networks, on the other hand, can easily represent functions with complex dependencies between inputs, and each model parameter is optimised with

Table 2. *Summary of systems evaluated; V denotes vocoded natural speech.*

System	Regression model	Regression target unit	Stream modelling	Variance	Duration-derived features	Enhancement method
V	-	-	-	-	-	-
D1	decision tree	state	separate	context-dependent	no	GV
D2	decision tree	state	separate	context-dependent	no	postfilter
N1	neural network	state	separate	context-dependent	no	postfilter
N2	neural network	state	separate	fixed	no	postfilter
N3	neural network	state	combined	fixed	no	postfilter
N4	neural network	frame	separate	fixed	no	postfilter
N5	neural network	frame	combined	fixed	no	postfilter
N6	neural network	frame	combined	fixed	yes	postfilter

regard to all training samples. By avoiding the hard partitioning of data that takes place in decision trees, the use of DNNs can mitigate the detrimental effects of across-linguistic-context averaging. It is certainly convincing that at least part of the preference for the DNN systems reported in [3] is due to the regression model used. However, it is unclear to what degree the other factors that varied between the HMM and DNN systems contributed to the reported preference.

2.2. Regression target unit

The use of subphonetic states which can stretch to account for variable numbers of acoustic observations is well motivated in ASR systems (including those typically used to force-align DNN training data) due to the large variability in duration observed between different instances of the same word or phone in human speech. From the speech generation point of view, however, the motivation is less clear. The statewise stationary approximation of speech is bound to blur some details of temporal evolution within states [1]. One technique proposed to address this problem in ASR is the use of *segmental HMMs* to explicitly model segment dynamics [9]. In TTS, it has been addressed by generating frame-level features directly [7, 3]. It is hypothesised that a contributing factor to the preference for the DNN system in [3] is the use of position-within-phone features. This paper seeks to confirm and quantify this contribution.

2.3. Combined vs. separate stream modelling

The source-filter model of speech assumes conditional independence of source and vocal-tract filter. The assumption is problematic because 1) there are in fact dependencies between F_0 and F_1 [10, 11], and 2) automatic techniques often attain poor separation of source and filter even given theoretical limitations (i.e. peaks at lower frequencies in the STRAIGHT spectrum tend to fit to harmonics rather than formants) [12]. [13] shows that it is possible to predict F_0 from MFCCs with considerable accuracy and confirms the correlation between source and vocal tract filter. It is possible that generating all acoustic parameters using a single deep neural network, as opposed to building separate regression models of each stream, can create more closely coupled parameter trajectories. In particular, modelling source and filter streams simultaneously can be viewed as a kind of multi-task learning, which has been found useful to improve naturalness [5, 14]. The preliminary study in [15] shows slight improvements when DNNs are trained to predict all streams simultaneously. In this investigation, the relative improvements gained from combined-stream modelling will be assessed.

2.4. Context-dependent vs. single fixed variance for MLPG

DNN-based systems generally use a fixed variance across all frames when maximum likelihood parameter generation (MLPG) is performed [3, 4, 5]. This differs from HMM systems, which learn context-dependent variances. In [16], no significant difference was found between a standard DNN system and a similar deep network (single-component MDN) which additionally predicted variances at the frame level. We also quantify the effect of switching a context-dependent variance for a fixed one by including a step to isolate this effect in our continuum of systems.

2.5. Use of duration features as inputs

The systems in [3, 5] used natural phone duration as a network input. If such systems are evaluated with oracle (i.e. forced-aligned) phone durations provided at synthesis time (like in [3, 5]), the strong correlations between variations in phone duration and e.g. F_0 excursions could enable a misleadingly large improvement which might not be possible to realise in practical applications, where natural durations are not available for synthesis. The present investigation verifies and quantifies the hypothesised increase in naturalness gained from the use of these extra duration-feature inputs, in the context of oracle durations.

3. SYSTEMS BUILT

3.1. Data

A database of speech from a British male speaker was used in the experiments, consisting of 2542 sentences: 2400 of these were used for training, 70 for validation and 72 for testing. The speech waveforms had been sampled at 48 kHz, and from them STRAIGHT [17] was used to extract 60-dimensional mel-cepstral coefficients (MCCs), 25 band aperiodicities (BAPs) and logarithmic fundamental frequency (log F_0) at 5 msec frame intervals.

3.2. Implementational details

Regression model type: Systems D1 and D2 are conventional HMM-based systems using decision trees to map from linguistic features to state-level distributions over acoustics. Tree-size was controlled using a minimum description length (MDL) criterion (penalty factor: 1.0). 2926 binary features were used as questions for node-splitting during the building of trees, although the effective size of the question set was considerably smaller than this as many of them were not capable of splitting the particular data-set which was used. The MCCs and BAPs with deltas and delta-deltas appended were modelled by single-component Gaussians, and $\log F_0$ with delta and delta-delta was modelled by a 3-dimensional multi-space probability distribution (MSD). The publicly available HTS toolkit [18] was used to implement the HMM systems.

All other systems (N1–N6) make use of DNNs to map from linguistic features to acoustics. The inputs to all neural net-based systems include 863 binary features which were a hand-selected subset of the decision tree systems' questions. This represents a slight inconsistency with systems D1 and D2: ideally, an additional step in the continuum of systems would control for this difference. Frame-aligned training data for the DNNs was obtained by forced alignment using the HMM system described above. DNN outputs consisted of MCCs, BAPs and linearly interpolated $\log F_0$ (all with deltas and delta-deltas) plus a voiced/unvoiced binary value. Input features were normalised to the range of [0.01, 0.99] and output features were standardised to have zero mean and unit variance. Note that D1–2 use MSD where N1–6 use interpolated F0: ideally this factor would have been isolated with an extra step in the continuum as this difference has been shown to have a significant effect on naturalness [19].

All networks trained have the same size and topology: 6 hidden layers with 1024 units in each. In all DNN systems $\tanh$ was used as the hidden unit activation function, and a linear activation function was employed at the output layer. No explicit regularisation was used during training. The mini-batch size was set to 256; a fixed learning rate and momentum were used. Momentum was 0.3, with a fixed learning rate chosen from among 5 rates (0.01, 0.003, 0.001, 0.0003, 0.0001) for each network on the basis of validation error. The learning rate of the top two layers was half that of other layers. The maximum number of epochs was set to 100. We implemented the networks using Theano version 0.6 [20] and training was done on a GPU.

State-level modelling: Systems D1 and D2 are conventional HMM-based systems making use of five-state, left-to-right hidden semi-Markov models (HSMM). Systems N1–3 were trained on the same number of frames as all other DNN systems; however, all frames, which had been aligned with the same HMM state during forced alignment shared exactly the same label, the 863 binary linguistic features being supplemented only with a single extra feature, the state index. In this way, the networks were trained to produce exactly the same prediction for all frames in a state, analogous with HMM state-level means (and in the case of system N1, variances). Systems N4–5 were trained with the 863 binary features supplemented with 2 extra input features: state index, and normalised position of frame within state. These 2 features are the bare minimum necessary for the networks to handle within-state variation over time. N6 used these 2 features along with several others (see below).

Separate-stream vs. combined-stream: D1–2 model streams independently in that separate decision trees are trained for the different streams. N1–3 also consisted of predictors trained independently for each stream: in each of these systems, 3 separate networks were trained separately to predict values for MCC, BAP and combined interpolated $\log F_0$ and voicing features. The learning rate was tuned separately for separate streams within a system, and all networks contained the same number of units. That is, no attempt was made to keep the number of model parameters constant across different models: the separate-stream systems contained approximately three times as many parameters as the combined-stream systems.

Fixed vs. context-dependent variance: MLPG was used by all systems using either pre-computed variances from the training data or predicted frame- or state-dependent variances to produce smooth and speech-like trajectories. D1–2 use the variances associated with each leaf of their decision trees in the standard way. In N2–6, the same technique as in [5] is used: the global standard deviation of the training corpus is used for every frame when generating parameters. System N1, in contrast, uses a single-component mixture density network to make a context-dependent prediction per frame: however, due to the way the inputs are coded, all frames in a given state will have the same predicted variance.

Full duration-derived features: System N6 supplements the 863 binary linguistic features with the full 9 duration-derived features we have used in previous work [5]. These include the two features used by systems N4 and N5, and consist of:

1. Fraction through state counting forwards
2. Fraction through state counting backwards
3. Fraction through phone counting forwards
4. Fraction through phone counting backwards
5. Position of state in phone counting forwards
6. Position of state in phone counting backwards
7. Length of state in frames
8. Length of phone in frames
9. Fraction of the current phone made up by current state

Note that due to normalisation by length, features 2 and 4 are redundant given features 1 and 3 respectively; similarly, because all phones contain 5 states, feature 6 is redundant given feature 5. Whilst this effective duplication means that more model parameters are dedicated to these features, we removed this duplication when coding the inputs for systems N1–5: for those systems, only items 1 and 5 were used.

Enhancement: During parameter generation, global variance (GV) enhancement was used by system D1; this was switched for postfiltering in D2 to allow fairer comparison with N1–6, whose output was also processed as is conventionally done in DNN systems with postfiltering. The Speech Signal Processing Toolkit (SPTK) was used to perform the same type of mel-cepstral domain formant emphasis as is used in the publicly available HTS demo script [18].

4. EVALUATION

4.1. Subjective test

To assess the subjective naturalness of synthetic speech produced by the different systems, we performed a MUSHRA (MUltiple Stimuli with Hidden Reference and Anchor) test [21]. This permits evaluation of multiple samples in a single trial without reducing the task to many pairwise comparisons. Specifically, subjects were instructed to rate a set of parallel, amplitude-normalised [22] stimuli on a scale from 0 (completely unnatural) to 100 (completely natural). Each set of stimuli represented the same sentence, but synthesised using all eight different systems in Section 3. Stimuli were ordered randomly and presented without labels. A reference stimulus containing matching vocoded natural speech was provided beside the rating sliders, and also included among the unlabelled examples, for a total of nine stimuli per set. Subjects were instructed to rate the hidden reference as completely natural, fixing the high end of the scale. No explicit lower anchor was used, since the synthetic speech stimuli themselves were sufficiently different from natural speech to act as implicit anchors. To familiarise the listeners with synthetic speech and with the rating interface, the test was preceded by a training phase and a set of tutorial sliders (not included in the analysis).

20 native English listeners, all students at the University of Edinburgh, participated in the test. Each listener rated two sets of ten synthesised Harvard sentences [23], with every set being approximately phonetically balanced. The two sentence sets were chosen from a larger pool of seven sets, balanced in such a way that each set was presented to at least five and at most six listeners. In total, 400 sets

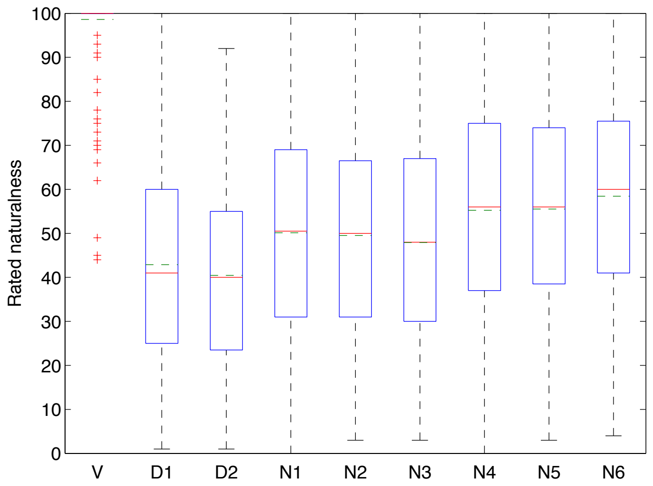

Fig. 1. Aggregated MUSHRA test results. Box edges are at 25 and 75% quantiles. Red line is the median, green dashed line is the mean.

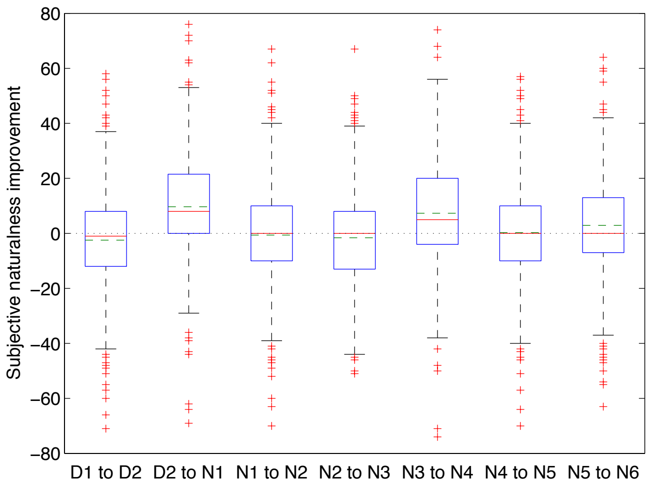

Fig. 2. Successive changes in naturalness for each step along the model continuum. Boxes and colours as in Figure 1.

of parallel ratings were obtained. All tests were conducted in sound-insulated booths over Beyerdynamic DT770 PRO headphones, with listeners being remunerated for their time and effort.

4.2. Results

Figure 1 presents the distribution of subjective ratings in a boxplot. It is clear that vocoded speech was judged as vastly more natural than the synthesised samples, though the distribution is quite broad due to variability between the different sentences and listeners. DNN-based systems outperform decision tree-based ones overall, though a few subjects had different preferences.

To gain further insight we performed double-sided pairwise Wilcoxon signed-rank comparisons between all systems, using the Holm-Bonferroni method to keep the familywise error rate below the level $\alpha = 0.05$. The analysis found no significant differences internally between systems in the following groups: {D1, D2}, {N1, N2, N3}, and {N4, N5}. All other systems, and all cross-group comparisons, are significantly different, statistically speaking. If each set of ratings is normalised to run from 0 to 100, then D1 also becomes significantly different from D2, but nothing else changes. Student's t-tests give the same conclusions as the Wilcoxon signed-rank tests in every single case. The statistically significant transitions thus occur when we switch:

1. Decision trees for neural networks (D2 to N1);
2. State-level modelling for frame-level modelling (N3 to N4);
3. Minimal frame-level features for the full set of durational features (N5 to N6).

The magnitudes of the change in rating from the first two of these are similar, while the third improvement is much smaller, as seen in Figure 2. The choice of whether to build separate or combined models for the different streams did not produce a significant difference in the evaluation, despite the fact that separate-stream systems have approximately three times as many parameters, which theoretically makes them more flexible but more difficult to optimise.

5. CONCLUSIONS

Based on the perceptual tests, we can conclude that the two changes with the greatest perceptual benefit in our experiment were *the switch from decision-tree to DNN regression*, and *the change from state- to frame-level targets for the regression*. Each of these netted an average gain close to 10 points on our 100-point naturalness scale. A much smaller but nonetheless statistically significant improvement was also observed from adding duration features to the input. Since the latter was achieved using oracle durations extracted from natural speech, it is unclear to what degree that gain be realised in practical synthesis systems without access to natural durations at run-time. The investigation in [24] uncovered no subjective difference between DNN-synthesised speech using DNN-predicted durations versus oracle durations for an audiobook corpus, but more expressive synthesis material with less predictable (but more prosodically informative) durations might exhibit a different pattern.

Other differences between typical HMM and DNN set-ups were not found to have a significant impact. In particular, using a globally fixed value for each parameter's variance during MLPG parameter generation was not harmful to synthesis output, confirming the finding of [16]. Contrary to [15], no significant improvement was observed when transitioning from a separate to a combined model of all streams. It is possible that the advantages depend on the corpus, and that materials with a greater F_0 range such as expressive speech or sung corpora might benefit more from models that enforce a close relationship between predicted feature values across streams.

The fact that two of the observed significant improvements were associated with an increase in the frame-level contextual information provided to the regression model is consistent with recent reports that wide-context stacked bottleneck features [5] or recurrent neural networks such as long-short term memory models (LSTMs) [25] further improve synthesis quality over standard DNN systems. While findings may vary with other corpora and system configurations, it appears clear that improved, frame-level regression techniques are likely to constitute a key component in high-quality statistical speech synthesis for the foreseeable future.

6. ACKNOWLEDGEMENTS

This research was supported by EPSRC Programme Grant EP/I031022/1, Natural Speech Technology (NST). The NST research data collection may be accessed at `http://hdl.handle.net/10283/786`. The test stimuli and response data for this paper can be accessed at `http://dx.doi.org/10.7488/ds/1316`.

7. REFERENCES

[1] Heiga Zen, Keiichi Tokuda, and Alan W. Black, "Statistical parametric speech synthesis," *Speech Communication*, vol. 51, no. 11, pp. 1039–1064, 2009.

[2] Thomas Merritt, Javier Latorre, and Simon King, "Attributing modelling errors in HMM synthesis by stepping gradually from natural to modelled speech," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, 2015, pp. 4220–4224.

[3] Heiga Zen, Andrew Senior, and Mike Schuster, "Statistical parametric speech synthesis using deep neural networks," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, 2013, pp. 7962–7966.

[4] Yao Qian, Yuchen Fan, Wenping Hu, and Frank K. Soong, "On the training aspects of deep neural network (DNN) for parametric TTS synthesis," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, 2014, pp. 3829–3833.

[5] Zhizheng Wu, Cassia Valentini-Botinhao, Oliver Watts, and Simon King, "Deep neural networks employing multi-task learning and stacked bottleneck features for speech synthesis," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, 2015, pp. 4460–4464.

[6] Kei Hashimoto, Keiichiro Oura, Yoshihiko Nankaku, and Keiichi Tokuda, "The effect of neural networks in statistical parametric speech synthesis," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, 2015, pp. 4455–4459.

[7] Alan W. Black, "CLUSTERGEN: a statistical parametric synthesizer using trajectory modeling," in *Proc. Interspeech*, 2006, pp. 1762–1765.

[8] Heiga Zen, "Acoustic modeling in statistical parametric speech synthesis – from HMM to LSTM-RNN," in *Proc. MLSLP*, 2015, Invited paper.

[9] W. J. Holmes and M. J. Russell, "Probabilistic-trajectory segmental HMMs," *Computer Speech & Language*, vol. 13, no. 1, pp. 3–37, 1999.

[10] Ann K. Syrdal and Shirley A. Steele, "Vowel F1 as a function of speaker fundamental frequency," *The Journal of the Acoustical Society of America*, vol. 78, no. S1, pp. S56, 1985.

[11] Gustav Eje Henter, Thomas Merritt, Matt Shannon, Catherine Mayo, and Simon 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.

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

[13] Ben Milner and Xu Shao, "Prediction of fundamental frequency and voicing from mel-frequency cepstral coefficients for unconstrained speech reconstruction," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 1, pp. 24–33, 2007.

[14] Qiong Hu, Zhizheng Wu, Korin Richmond, Junichi Yamagishi, Yannis Stylianou, and Ranniery Maia, "Fusion of multiple parameterisations for DNN-based sinusoidal speech synthesis with multi-task learning," in *Proc. Interspeech*, 2015, pp. 854–858.

[15] Bo Chen, Zhehuai Chen, Jiachen Xu, and Kai Yu, "An investigation of context clustering for statistical speech synthesis with deep neural network," in *Proc. Interspeech*, 2015, pp. 2212–2216.

[16] Heiga Zen and Andrew Senior, "Deep mixture density networks for acoustic modeling in statistical parametric speech synthesis," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, 2014, pp. 3872–3876.

[17] Hideki Kawahara, Ikuyo Masuda-Katsuse, and Alain de Cheveigné, "Restructuring speech representations using a pitch-adaptive time–frequency smoothing and an instantaneous-frequency-based F0 extraction: Possible role of a repetitive structure in sounds," *Speech Communication*, vol. 27, no. 3, pp. 187–207, 1999.

[18] Heiga Zen, Takashi Nose, Junichi Yamagishi, Shinji Sako, Takashi Masuko, Alan W. Black, and Keiichi Tokuda, "The HMM-based speech synthesis system (HTS) version 2.0," in *Proc. ISCA SSW6*, 2007, pp. 294–299.

[19] Zhehuai Chen and Kai Yu, "An investigation of implementation and performance analysis of DNN based speech synthesis system," in *Proc. IEEE 12th Int. Conference on Signal Processing*, 2014, pp. 577–582.

[20] James Bergstra, Olivier Breuleux, Frédéric Bastien, Pascal Lamblin, Razvan Pascanu, Guillaume Desjardins, Joseph Turian, David Warde-Farley, and Yoshua Bengio, "Theano: a CPU and GPU math expression compiler," in *Proc. Python for Scientific Computing Conference (SciPy)*, 2010.

[21] International Telecommunication Union Radiocommunication Assembly, Geneva, Switzerland, *Method for the subjective assessment of intermediate quality level of audio systems*, June 2014.

[22] International Telecommunication Union, Telecommunication Standardization Sector, Geneva, Switzerland, *Objective measurement of active speech level*, March 2011.

[23] E. H. Rothauser, W. D. Chapman, N. Guttman, K. S. Nordby, H. R. Silbiger, G. E. Urbanek, and M. Weinstock, "IEEE recommended practice for speech quality measurements," *IEEE Transactions on Audio and Electroacoustics*, vol. 17, no. 3, pp. 225–246, 1969.

[24] Gustav Eje Henter, Srikanth Ronanki, Oliver Watts, Mirjam Wester, Zhizheng Wu, and Simon King, "Robust TTS duration modelling using DNNs," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, 2016.

[25] Yuchen Fan, Yao Qian, Fenglong Xie, and Frank K. Soong, "TTS synthesis with bidirectional LSTM based recurrent neural networks," in *Proc. Interspeech*, 2014, pp. 1964–1968.