Uncertainty propagation through deep neural networks

Ahmed Hussen Abdelaziz, Shinji Watanabe, John Hershey, Emmanuel Vincent, Dorothea Kolossa

Uncertainty Propagation through Deep Neural Networks

Ahmed Hussen Abdelaziz[1*], *Shinji Watanabe*[2], *John R. Hershey*[2],
Emmanuel Vincent[3], *Dorothea Kolossa*[1]

[1] Cognitive Signal Processing Group, Institute of Communication Acoustics,
Ruhr-Universität Bochum, Germany
[2] Mitsubishi Electric Research Laboratories, Cambridge, MA, USA
[3] Inria, Villers-lès-Nancy, France
{ahmed.hussenabdelaziz, dorothea.kolossa}@rub.de, {watanabe, hershey}@merl.com,
emmanuel.vincent@inria.fr

Abstract

In order to improve the ASR performance in noisy environments, distorted speech is typically pre-processed by a speech enhancement algorithm, which usually results in a speech estimate containing residual noise and distortion. We may also have some measures of uncertainty or variance of the estimate. Uncertainty decoding is a framework that utilizes this knowledge of uncertainty in the input features during acoustic model scoring. Such frameworks have been well explored for traditional probabilistic models, but their optimal use for deep neural network (DNN)-based ASR systems is not yet clear. In this paper, we study the propagation of observation uncertainties through the layers of a DNN-based acoustic model. Since this is intractable due to the nonlinearities of the DNN, we employ approximate propagation methods, including Monte Carlo sampling, the unscented transform, and the piecewise exponential approximation of the activation function, to estimate the distribution of acoustic scores. Finally, the expected value of the acoustic score distribution is used for decoding, which is shown to further improve the ASR accuracy on the CHiME database, relative to a highly optimized DNN baseline.

Index Terms: Noise-robust ASR, Deep Neural Networks, Observation Uncertainty, Uncertainty Propagation

1. Introduction

Although statistical models like hidden Markov models (HMMs) have shown great success in modeling and recognizing the temporal evolution of the spectral characteristics of speech, wider use of automatic speech recognition (ASR) systems is still precluded by acoustic environmental effects like background noise and reverberation. In order to achieve an acceptable level of recognition robustness against these effects, the distorted speech is usually de-noised using speech enhancement algorithms [1, 2]. However, the enhanced signals obtained from such algorithms are not perfectly compensated and often contain residual noise, estimation errors, and even artifacts introduced by the speech enhancement algorithms.

Modeling the spectral characteristics of speech signals using statistical models like GMMs has facilitated the development of a range of so-called probabilistic uncertainty-of-observation (UoO) techniques [3–7]. Such techniques take into account the residual noise and the estimation errors of the enhancement algorithms by considering the enhanced speech outputs as random variables rather than point estimates. In GMMs, one can easily take the uncertainty into account for Gaussian observations by marginalizing out the random observation variables.

In DNN-based systems, in contrast, the observations are not explicitly modeled by statistical distributions. Although there are probabilistic interpretations of DNNs, here we consider standard sigmoid DNNs simply as deterministic nonlinear functions. To incorporate the uncertainty for Gaussian-distributed observations, we have to solve two problems: first, compute the distribution of acoustic scores by integrating out the input random variables, and second, incorporate the score distribution into the decoding algorithm once it has been estimated. Unfortunately, computing the score distribution requires integrating the DNN over all input values, according to the observation distribution. However, due to the nonlinearities, this integral is intractable for DNNs and requires approximation. The problem of decoding is complicated by its dependency on the approximations that are used to estimate the score distribution.

We therefore investigate both problems together. For estimating score distribution, we investigate Monte-Carlo methods, including Monte Carlo sampling, the unscented transform, and the piecewise exponential approximation of the activation function. For decoding, we investigate different methods of integrating over score distributions, where we take two expected values related to the posterior state probabilities.

In [8], uncertainty propagation has been conducted layer by layer and for each node separately by approximating the nonlinear activation function using the so-called piece-wise exponential (PIE) approximation or by approximating the input Gaussian distribution using the unscented transform (UT). These approximations require the hidden layer pre-activations to be statistically independent. In this paper, we show by Monte Carlo simulation that this condition is not always true, especially for deep and wide DNNs. In order to minimize the estimation errors and to avoid the accumulation of the propagated errors from a hidden layer to another, we propose to use Monte Carlo sampling and the UT with the entire DNN. This allows the propagation of uncertainty even through the softmax layer, which is difficult using layer-wise approximation methods.

In [8], the acoustic scores have been modified to match the observation uncertainties in a similar way as used in the uncertainty decoding approach [4], which has first been introduced for GMM-based ASR systems. In this paper, we investigate

* This work was conducted while the first author was doing an internship at Mitsubishi Electric Research Laboratories.

an alternative approach to exploit the uncertainties for acoustic scoring by replacing the DNN pseudo log-likelihoods by their conditional expectations given the enhanced features. This new score is reminiscent of the GMM-based modified acoustic scores in [9]. This approach can be used in conjunction with the layer-wise uncertainty propagation methods as it does not require propagation through the softmax layer.
The remaining paper is organized as follows: In Section 2, Monte Carlo sampling, the unscented transform, and the PIE approximation are described as possible approaches for uncertainty propagation through DNNs. Next, in Section 3, the two uncertainty-based acoustic scores are introduced. In Section 4, all approaches are evaluated using the second track of the second CHiME challenge [14]. Finally, in Section 5, the paper is concluded and an outlook of future work is given.

2. Uncertainty Propagation through DNNs

DNN layers are composed of a linear operation followed by a nonlinear operation. The typically used nonlinear function is a sigmoid function in the hidden layers and a softmax function in the output layer. The question to be addressed in this section is the following. If the input to a DNN is a multivariate Gaussian random variable, what is the distribution of the corresponding output random variable after applying the linear and nonlinear operations of all neural network layers?

2.1. Uncertainty Propagation through Entire DNN

2.1.1. Monte Carlo Sampling

Monte Carlo sampling is the simplest approach that can be used to calculate the statistics of random variables that undergo a nonlinear transformation. This method is based on randomly drawing a number of samples from the distribution underlying the random variable. The nonlinear transformation is then applied to these samples. The first and second order statistics of the nonlinearly transformed random variable can be estimated as the mean and the variance of the output samples, respectively.

2.1.2. Unscented Transform

The UT is a similar method to the Monte Carlo approach. However, in the UT, the samples are not drawn randomly but according to a specific criterion. For an I-dimensional random variable, $2I+1$ sample vectors and their associated weights are computed as introduced in [10]. The nonlinear function, here the DNN, is then applied to these sample vectors. Using the estimated weights, the first and second order statistics of the output distribution are calculated as a weighted sum of the output sample vectors and a weighted sum of the squared mean-free samples, respectively.

2.2. Layer-Wise Uncertainty Propagation

Instead of propagating the uncertainty through the entire DNN at once, the uncertainties can be propagated layer by layer. Propagating a multi-variate Gaussian distribution through the linear part of a neural network layer is simple, as another Gaussian distribution is analytically obtained. On the other hand, propagating the multi-variate Gaussian distribution through a sigmoid function results in a very complex distribution as shown in [11]. This has led the authors of [8] to use simpler approximations like the PIE approximation [12] and the UT to determine the first and second order statistics of this complex distribution for every layer.

2.2.1. PIE Approximation

In the PIE approximation, a sigmoid function $g(z)$ is approximated by a sum of two exponential functions as follows:

$$g(z) = \frac{1}{1+e^{-z}} \approx 2^{z-1}u(-z) + (1-2^{(-z-1)})u(z), \quad (1)$$

where $u(z)$ is the unit step function. If z is a one-dimensional Gaussian random variable with mean value μ_z and standard deviation σ_z, the first and second order statistics of (1) can be estimated as follows:

$$\begin{aligned}\mathbb{E}[g(z)] = {} & 2^{\left(\mu_z + 0.5\log(2)\sigma_z^2 - 1\right)}\phi\left(-\frac{\mu_z}{\sigma_z} - \log(2)\sigma_z\right) \\ & - 2^{\left(-\mu_z + 0.5\log(2)\sigma_z^2 - 1\right)}\left[1 - \phi\left(-\frac{\mu_z}{\sigma_z} + \log(2)\sigma_z\right)\right] \\ & + \left[1 - \phi\left(-\frac{\mu_z}{\sigma_z}\right)\right] \end{aligned} \quad (2)$$

$$\text{Var}\left[g(z)\right] = \mathbb{E}\left[g(z)^2\right] - \left(\mathbb{E}[g(z)]\right)^2, \quad (3)$$

where

$$\begin{aligned}\mathbb{E}\left[g(z)^2\right] = {} & 2^{\left(2\mu_z + 2\log(2)\sigma_z^2 - 2\right)}\phi\left(-\frac{\mu_z}{\sigma_z} - 2\log(2)\sigma_z\right) \\ & - 2^{\left(-\mu_z + 0.5\log(2)\sigma_z^2\right)}\left[1 - \phi\left(-\frac{\mu_z}{\sigma_z} + \log(2)\sigma_z\right)\right] \\ & + 2^{\left(-2\mu_z + 2\log(2)\sigma_z^2 - 2\right)}\left[1 - \phi\left(-\frac{\mu_z}{\sigma_z} + 2\log(2)\sigma_z\right)\right] \\ & + 1 - \phi\left(-\frac{\mu_z}{\sigma_z}\right). \end{aligned} \quad (4)$$

In (2) and (4), ϕ is the cumulative density function of the standard normal distribution.

2.3. Discussion

For very wide and deep DNNs, propagating a Gaussian distribution through the sigmoid function of a hidden layer using the UT or the PIE approximation is computationally very expensive. For example, deploying the UT needs 4097 vectors and their associated weights to be computed, where the dimension of the hidden layers used in this study is I =2048. It is also difficult to use the PIE approximation, since the computation of the cumulative density function for multi-variate Gaussian distributions of such a large dimension is not trivial when the covariance matrix is not diagonal. Therefore, the off-diagonal components of the covariance matrix of the hidden layer pre-activations have been neglected in [8] assuming a weak correlation between their components. Based on this assumption, the PIE approximation and the UT have been applied for each neuron separately.
The weak correlation assumption demands a diagonal covariance matrix of the the pre-activations, which can be computed for the n^{th} layer via

$$\left[\mathbf{\Sigma}_{\mathbf{z}^n}\right]_{i,j} = \sum_{k=1}^{I}\sum_{k'=1}^{I}\left[\mathbf{W}^n\right]_{i,k}\ \left[\mathbf{W}^n\right]_{j,k'}\ \left[\mathbf{\Sigma}_{\mathbf{h}^{(n-1)}}\right]_{k,k'}. \quad (5)$$

In (5), $\mathbf{W}^n$ and $\mathbf{\Sigma}_{\mathbf{h}^{(n-1)}}$ are the weight matrix of the n^{th} hidden layer and the covariance matrix of the preceding hidden layer, respectively. From (5), the covariance matrix $\mathbf{\Sigma}_{\mathbf{z}^n}$ becomes diagonal if the weight matrix $\mathbf{W}^n$ and the covariance

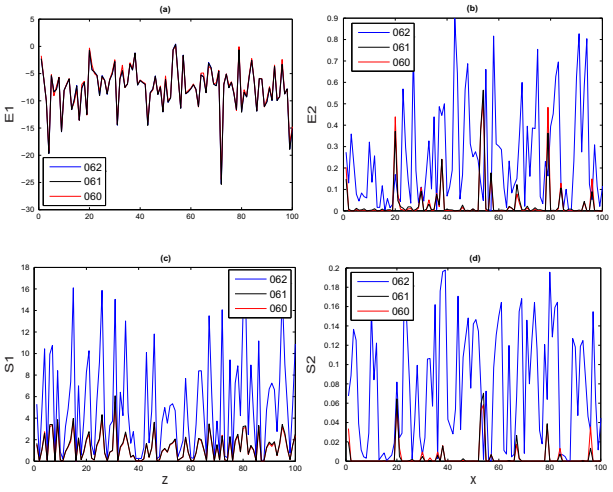

Figure 1: (a), (b) Estimated mean vector and covariance matrix diagonal of the second hidden layer pre-activations, respectively. (c), (d) Estimated mean vector and covariance matrix diagonal of the second hidden layer, respectively.

matrix $\mathbf{h}^{(n-1)}$ are also diagonal. Although the off-diagonal components of the covariance matrix $\mathbf{\Sigma}_{\mathbf{h}^{(n-1)}}$ are small indicating weak correlation between the components of $\mathbf{h}^{(n-1)}$, see, e.g., [13], this is not the case for the weight matrix $\mathbf{W}^n$. On the contrary, the redundancy of neurons at deeper layers may cause more correlation between the components of $\mathbf{z}^n$.

Moreover, the diagonal components of $\mathbf{\Sigma}_{\mathbf{z}^n}$ also suffer from estimation errors as the weak correlation between the components of $\mathbf{h}^{(n-1)}$ is compensated by the large number of summation terms in (5). Because of the nonlinearity, the estimation errors of the covariance matrix $\mathbf{\Sigma}_{\mathbf{z}^n}$ cause further estimation errors of both the mean vector and the covariance matrix of the corresponding hidden layer output $\mathbf{h}^n$. In order to give a concrete example of this phenomenon, a Monte Carlo simulation is conducted as follows. One acoustic feature vector has been extracted from one arbitrary utterance of the CHiME corpus [14]. This feature vector has been considered as the mean vector of a multivariate Gaussian distribution. The covariance matrix of this Gaussian distribution has been assumed to be diagonal with randomly chosen diagonal entries. About 10,000 samples have been drawn from this Gaussian distribution and then processed by a pre-trained DNN[1]. The ground-truth first and second order statistics of the propagated distribution at a hidden layer have been estimated as the mean and the covariance of the samples propagated through this layer. These parameters are always referred to as the Monte-Carlo (MC) estimated parameters.

Fig. 1 shows part of the mean vector and the diagonal of the covariance matrix of the second hidden layer and its pre-activation. As can be seen, the diagonal components of the covariance matrix $\mathbf{\Sigma}_{\mathbf{z}^2}$ are much smaller than the true values. The estimation errors of $\mathbf{\Sigma}_{\mathbf{z}^2}$ are reflected in further errors in the estimated statistics of the hidden layer outputs $\mathbf{h}^2$.

In order to reduce the estimation errors and avoid their propagation through the DNN, Monte Carlo sampling can be used to estimate the distribution of the scores when the posterior uncertainty is propagated. However, the accuracy of this method depends on the number of samples, namely, the more samples are drawn from the input distribution, the more accurate the estimation of the statistics in the likelihood domain becomes. On the other hand, increasing the number of samples may reduce the computational efficiency of this method. Finally, the UT can also be used with the entire DNN as an alternative deterministic approach to Monte Carlo sampling.

[1] More details about the used DNN and its training are introduced in Section 4.

3. DNN-based Decoding of Uncertain Data

In DNN-based ASR, decoding is carried out using pseudo log-likelihoods instead of real log-likelihoods. For a DNN of L hidden layers, the pseudo log-likelihood of a clean feature vector $\mathbf{X}$ given a state q_i is usually estimated via

$$\mathcal{L}^{\text{pseudo}} = \log\left(\mathrm{p}_{\text{pseudo}}(\mathbf{X}|q_i)\right) = \mathbf{z}_i^{L+1} - \log\left(\mathrm{p}(q_i)\right). \quad (6)$$

In (6), $\mathbf{z}_i^{L+1}$ is the i^{th} pre-activation of the DNN output layer, where $i \in \{1, \ldots, I\}$, and $\mathrm{p}(q_i)$ is the prior probability of the i^{th} state q_i. There are actually two missing terms in (6) that distinguish $\mathcal{L}^{\text{pseudo}}$ from the conventional log-likelihood acoustic score, which can be estimated via

$$\log\left(p(\mathbf{X}|q_i)\right) = \log\left(\frac{\mathrm{p}(q_i|\mathbf{X})\mathrm{p}(\mathbf{X})}{\mathrm{p}(q_i)}\right) \quad (7)$$

$$= \mathbf{z}_i^{L+1} - \log \mathrm{p}(q_i) + \log\left(\mathrm{p}(\mathbf{X})\right) - \log\left(\sum_{j=1}^{I} \exp\left(\mathbf{z}_j^{L+1}\right)\right). \quad (8)$$

As can be seen in (8), the two missing terms are the logarithm of the softmax normalization constant $\log\left(\sum_{j=1}^{I} \exp\left(\mathbf{z}_j^{L+1}\right)\right)$ and the clean feature log-prior $\log\left(\mathrm{p}(\mathbf{X})\right)$. Since these terms are constants for all states, the decoding procedure is not affected by replacing the log-likelihood in (8) by the pseudo log-likelihood in (6).

For uncertain data, the posterior probability $\mathrm{p}(\mathbf{X}|\mathbf{Y})$ of a clean feature vector $\mathbf{X}$ given an enhanced feature vector $\mathbf{Y}$ is observed instead of the clean features. One possibility to deploy this posterior in the decoding procedure is to replace the pseudo log-likelihood in (6) by its conditional expectation given the enhanced features, which yields the following acoustic score

$$\mathcal{L}^{(\text{OU1})} = \mathbb{E}\left[\log\left(\mathrm{p}_{\text{pseudo}}(\mathbf{X}|q_i)\right)|\mathbf{Y}\right] \quad (9)$$

$$= \mathbb{E}\left[\mathbf{z}_i^{L+1}|\mathbf{Y}\right] - \log\left(\mathrm{p}(q_i)\right). \quad (10)$$

In order to estimate the expectation $\mathbb{E}\left[\mathbf{z}_i^{L+1}|\mathbf{Y}\right]$, the posterior $\mathrm{p}(\mathbf{X}|\mathbf{Y})$ should first be propagated up to the pre-activations of the output layer. The mean value of the propagated distribution can then be deployed in (10) as an estimate of $\mathbb{E}\left[\mathbf{z}_i^{L+1}|\mathbf{Y}\right]$.

In [8], another modified score has been obtained by replacing the clean posterior $p(q_i|\mathbf{X})$ by the enhanced posterior $p(q_i|\mathbf{Y})$, which can be found by integrating the joint probability $p(q_i, \mathbf{X}|\mathbf{Y})$ over the space of the clean features as follows:

$$p(q_i|\mathbf{Y}) = \int_{\mathbf{X}} p(q_i, \mathbf{X}|\mathbf{Y}) d\mathbf{X} \quad (11)$$

$$= \mathbb{E}\left[p(q_i|\mathbf{X})|\mathbf{Y}\right] = \mathbb{E}\left[\mathbf{h}_i^{L+1}|\mathbf{Y}\right], \quad (12)$$

where $\mathbf{h}_i^{L+1}$ is the output of the softmax layer. The transition from (11) to (12) assumes statistical independence of the state q_i and the enhanced features $\mathbf{Y}$ given the clean features $\mathbf{X}$. Using the enhanced posterior in (12), the acoustic score can be estimated as follows:

$$\mathcal{L}^{(\text{OU2})} = \log\left(p(q_i|\mathbf{Y})\right) - \log\left(\mathrm{p}(q_i)\right). \quad (13)$$

It can be seen in (12) that the posterior $\mathrm{p}(\mathbf{X}|\mathbf{Y})$ should be propagated to the DNN output layer in order to estimate the enhanced posteriors $p(q_i|\mathbf{Y})$ and hence, the new acoustic score $\mathcal{L}^{(\mathrm{UD2})}$. On the other hand, the modified acoustic score in (10) does not need the posterior $\mathrm{p}(\mathbf{X}|\mathbf{Y})$ to be propagated through the softmax function of the output layer, which makes it suitable for the layer-wise uncertainty propagation approaches.

4. Experiments and Results

4.1. Dataset

Track 2 of the 2nd CHiME Challenge [14] has been used for evaluation. The task is to recognize English sentences read by different male and female speakers taken from the medium vocabulary subset (5,000 words) of the Wall Street Journal (WSJ0) corpus [15]. The training dataset contains 7138 noisy utterances spoken by 84 speakers. The development and the test dataset contain 2454 and 1980 noisy utterances, respectively. The noisy utterances have been created by first convolving the WSJ0 clean utterances with binaural room impulse responses (BRIRs) and then adding background noise signals at six different SNRs: -6, -3, 0, 3, 6, and 9 dB. The BRIRs and the background noise signals have been recorded in a domestic living room using a head and torso simulator (HATS).

4.2. Experimental Setup

The baseline ASR system has been trained as follows [16, 17]. The DNN target state posteriors have been estimated using pre-trained triphone GMM models. Training of the GMM models has been done using the clean signals underlying the noisy training and development utterances. The features used for training are the 13 MFCC features with their corresponding Δ and $\Delta\Delta$ features. The MFCC features have been post-processed using linear discriminant analysis (LDA) [18, 19], maximum likelihood linear regression (MLLR) [20, 21], and speaker adaptive training (SAT) [22].

The DNN input features have been computed as follows. First, the noisy signals have been enhanced using a multichannel NMF pre-processor [23, 24]. From the enhanced signals, 40-dimensional Mel feature vectors have been obtained. Finally, 11 frames (5 previous frames, current frame, and 5 following frames) have been appended to form the DNN input features.

The DNN is composed of a 440-dimensional input layer, seven 2048-dimensional hidden layers, and an output layer of 2004 nodes representing the HMM states. The parameters of the hidden layers have been initialized using restricted Boltzmann machine (RBM) pre-training. Finally, all parameters have been fine-tuned using the back-propagation algorithm [25].

The ASR performance has been evaluated in terms of the word error rate (WER). Training and decoding have been conducted using the Kaldi speech recognition toolkit [26]. The modified acoustic scores (10) and (13) and the uncertainty propagation approaches have been applied using the DNN uncertainty propagation toolbox [27].

4.3. Results

The ground-truth results have been obtained using a Monte Carlo simulation, which has been conducted as follows. Each 440-dimensional DNN input feature vector has been considered as the mean vector of the Gaussian distribution $\mathrm{p}(\mathbf{X}|\mathbf{Y})$. Similarly to [28], the covariance matrix of $\mathrm{p}(\mathbf{X}|\mathbf{Y})$ has been assumed to be diagonal with diagonal entries defined as the squared difference between the corresponding components of the noisy and the enhanced features weighted by a dimension-, state-, and SNR-independent constant η. A grid search with minimum WER criterion has been conducted using the development set to find the appropriate values of η. $\eta = 0.3$ and $\eta = 0.4$ have achieved the best results using the acoustic scores (10) and (13), respectively. The same estimates of the mean vector and the covariance matrix of $\mathrm{p}(\mathbf{X}|\mathbf{Y})$ have also been used with the PIE approximation, the layer-wise UT, and the entire-DNN UT. From the Gaussian distribution $\mathrm{p}(\mathbf{X}|\mathbf{Y})$, 50 vectors have been sampled and applied to the DNN. The sample mean of the 50 corresponding DNN output vectors $\mathbf{h}^{L+1}$ and their pre-activations $\mathbf{z}^{L+1}$ have been considered as estimates of the mean vectors $\mathbb{E}\left[\mathbf{h}^{L+1}|\mathbf{Y}\right]$ and $\mathbb{E}\left[\mathbf{z}^{L+1}|\mathbf{Y}\right]$.

For the sake of simplicity and in order to make the entire-DNN UT more practical, we have used just three samples instead of the $(2I+1)$ required samples. The three samples are simply the mean vector of the posterior $\mathrm{p}(\mathbf{X}|\mathbf{Y})$ and the mean vector plus/minus $\sqrt{3}$ times the covariance matrix diagonal.

In Table 1, we compare the recognition results of the development and test set obtained using the conventional pseudo log-likelihood $\mathcal{L}^{(\mathrm{pseudo})}$ and the uncertainty-based acoustic scores $\mathcal{L}^{(\mathrm{OU1})}$ and $\mathcal{L}^{(\mathrm{OU2})}$. Since the PIE approximation and the layer-wise UT can only be used for propagating the uncertainty to the output layer pre-activations, these approaches can only be tested with the acoustic score $\mathcal{L}^{(\mathrm{OU1})}$. As can be seen, the best results are achieved using Monte Carlo sampling. It can also be noticed that the results obtained using the acoustic score $\mathcal{L}^{(\mathrm{OU2})}$ are better than those results achieved using $\mathcal{L}^{(\mathrm{OU1})}$. Despite the small number of samples used in the entire-DNN UT, it gives better results than those obtained using the PIE approximation and the unscented transform, while approaching the results achieved using Monte Carlo sampling.

Table 1: WER results obtained using different acoustic scores and different uncertainty propagation (UP) approaches.

Acoustic Score	UP	Dev.	Test
$\mathcal{L}^{(\mathrm{pseudo})}$ (Baseline)	—	27.59	21.67
$\mathcal{L}^{(\mathrm{OU1})}$	PIE	31.80	23.82
	Layer-wise UT	28.55	21.78
	Entire-DNN UT	27.80	21.35
	MC	27.27	21.27
$\mathcal{L}^{(\mathrm{OU2})}$	Entire-DNN UT	**26.88**	21.23
	MC	27.03	**21.06**

5. Conclusions

In this paper, four possible approaches of uncertainty propagation through DNNs have been investigated: Monte Carlo sampling, entire-DNN unscented transform, PIE approximation, and layer-wise unscented transform. The propagated uncertainties have been deployed in the DNN decoding procedure using two modified acoustic scores. The best results have been achieved using Monte Carlo sampling for uncertainty propagation and the second modified acoustic score, which is similar to the uncertainty decoding approach originally proposed for GMMs in [9]. As an alternative practical approach with a greatly reduced computational effort, the entire-DNN UT can also be used for uncertainty propagation.

This work can be extended by improving the uncertainty estimation using dynamic SNR-dependent weighting constants η instead of the static SNR-independent ones used in this study.

6. References

[1] J. Li, L. Deng, Y. Gong, and R. Haeb-Umbach, "An overview of noise-robust automatic speech recognition," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 22, no. 4, pp. 745–777, 2014.

[2] D. Kolossa and R. Haeb-Umbach, *Robust speech recognition of uncertain or missing data*. Springer, 2011.

[3] J. A. Arrowood and M. A. Clements, "Using observation uncertainty in HMM decoding," in *ICSLP*, Denver, Colorado, Sepember 2002.

[4] L. Deng, J. Droppo, and A. Acero, "Dynamic compensation of HMM variances using the feature enhancement uncertainty computed from a parametric model of speech distortion," *IEEE Transactions on Speech and Audio Processing*, vol. 13, no. 3, pp. 412–421, 2005.

[5] A. H. Abdelaziz, S. Zeiler, D. Kolossa, V. Leutnant, and R. Haeb-Umbach, "GMM-based significance decoding," in *ICASSP*, Vancouver, Canada, 2013.

[6] D. Kolossa, S. Zeiler, R. Saeidi, and R. Astudillo, "Noise-adaptive LDA: A new approach for speech recognition under observation uncertainty," *IEEE Signal Processing Letters*, vol. 20, no. 11, pp. 1018–1021, 2013.

[7] D. T. Tran, E. Vincent, and D. Jouvet, "Fusion of multiple uncertainty estimators and propagators for noise robust ASR," in *ICASSP*, Florence, Italy, 2014.

[8] R. F. Astudillo and J. P. da Silva Neto, "Propagation of uncertainty through multilayer perceptrons for robust automatic speech recognition," in *Interspeech*, Florence, Italy, 2011.

[9] A. Ozerov, M. Lagrange, and E. Vincent, "GMM-based classification from noisy features," in *International Workshop on Machine Listening in Multisource Environments (CHiME)*, Florence, Italy, 2011.

[10] S. Julier and J. Uhlmann, "A new extension of the Kalman filter to nonlinear systems," in *11th International Symposium on Aerospace/Defence Sensing, Simulation and Controls*, 1997.

[11] Y. Lee and S.-H. Oh, "Input noise immunity of multilayer perceptrons," *ETRI*, vol. 16, pp. 35–43, 1994.

[12] V. Beiu, J. A. Peperstrate, J. Vandewalle, and R. Lauwereins, "VLSI complexity reduction by piece-wise approximation of the sigmoid function," in *EANN*, Brussels, Belgium, 1994.

[13] S.-H. Oh and Y. Lee, "Effect of nonlinear transformations on correlation between weighted sums in multilayer perceptrons," *IEEE Transactions on Neural Networks*, vol. 5, no. 3, pp. 508–510, 1994.

[14] E. Vincent, J. Barker, S. Watanabe, J. Le Roux, F. Nesta, and M. Matassoni, "The second CHiME speech separation and recognition challenge: Datasets, tasks and baselines," in *ICASSP*, Vancouver, Canada, 2013.

[15] J. Garofalo, D. Graff, D. Paul, and D. Pallett, "CSR-I (WSJ0) complete," in *Linguistic Data Consortium*, Philadelphia, USA, 2007.

[16] Y. Tachioka, S. Watanabe, J. L. Roux, and J. R. Hershey, "Discriminative methods for noise robust speech recognition: A CHiME challenge benchmark," in *The 2nd International Workshop on Machine Listening in Multisource Environments (CHiME)*, Vancouver, Canada, 2013.

[17] C. Weng, D. Yu, S. Watanabe, and B.-H. F. Juang, "Recurrent deep neural networks for robust speech recognition," in *ICASSP*, Florence, Italy, 2014.

[18] R. Haeb-Umbach and H. Ney, "Linear discriminant analysis for improved large vocabulary continuous speech recognition," in *ICASSP*, vol. 1, 1992, pp. 13–16.

[19] C. Avendano, S. van Vuuren, and H. Hermansky, "Data-based RASTA-like filter design for channel normalization in ASR," in *ICSLP*, Philadelphia, USA, 1996.

[20] C. Leggetter and P. Woodland, "Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models," *Computer Speech & Language*, vol. 9, no. 2, pp. 171 – 185, 1995.

[21] M. J. F. Gales and P. C. Woodland, "Mean and variance adaptation within the MLLR framework," *Computer Speech and Language*, vol. 10, pp. 249–264, 1996.

[22] T. Anastasakos, J. McDonough, R. Schwartz, and J. Makhoul, "A compact model for speaker-adaptive training," in *ICSLP*, Philadelphia, USA, 1996.

[23] A. Ozerov, E. Vincent, and F. Bimbot, "A general flexible framework for the handling of prior information in audio source separation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 4, pp. 1118 – 1133, May 2012.

[24] Y. Salaün, E. Vincent, N. Bertin, N. Souviraà-Labastie, X. Jaureguiberry, D. T. Tran, and F. Bimbot, "The flexible audio source separation toolbox version 2.0," in *ICASSP Show & Tell*, Florence, Italy, 2014.

[25] G. Hinton, L. Deng, D. Yu, A. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. S. G. Dahl, and B. Kingsbury, "Deep neural networks for acoustic modeling in speech recognition," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97, November 2012.

[26] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlíček, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, and K. Vesely, "The Kaldi speech recognition toolkit," in *IEEE Workshop on Automatic Speech Recognition and Understanding*, Big Island, Hawaii, US, 2011.

[27] A. H. Abdelaziz and S. Watanabe, "Kaldi/Matlab DNN-based uncertainty propagation tools," retrieved February 2015. [Online]. Available: https://github.com/makladios/Kaldi_Matlab_DNN_UP

[28] M. Delcroix, S. Watanabe, T. Nakatani, and A. Nakamura, "Cluster-based dynamic variance adabtation for interconnecting speech enhancement pre-processor and speech recognizer," *Computer Speech & Language*, vol. 27, no. 1, pp. 350–368, 2013.