Improving generalization of Monte Carlo dropout based DNN ensemble model for speech enhancement and results on real world, traffic noise

Nazreen P.M.
Department of Electrical Engineering
Indian Institute of Science
Bangalore, India, 560012
Email: nazreenp@iisc.ac.in

A.G. Ramakrishnan
Department of Electrical Engineering
Indian Institute of Science
Bangalore, India, 560012
Email: agr@iisc.ac.in

***Abstract*—We propose a threshold-based algorithm to choose between model uncertainty-based and DNN-classifier-based selection of noise-specific DNN models for speech enhancement, using Monte Carlo dropout. This method tries to compensate for the poor performance of the former scheme on speech with seen noises compared to classifier-based scheme. We show some promising results on speech corrupted with a mixture of unseen noises and on time varying, non-stationary noises, affecting random segments of speech. We use TIMIT speech, NOISEX-92 noises, and real world, traffic noise recorded by us. Our algorithm performs well on real world, traffic noise from 10 down to -10 dB.**

I. Introduction

Deep learning [1], [2] based speech enhancement techniques are being widely used recently because of the ability of DNNs to learn any complex functions [3]. One of the major drawbacks of DNN based enhancement is its inability to perform well for an unseen noise scenario, that is, the case where the network is less adapted to the noise that affects the input speech. One way to address this issue is to train the DNN model using a variety of acoustic conditions [4], [5], [6].

Model-specific enhancement techniques [7], [8], [9], [10] have gained popularity recently which depend on a model selector, which ensures that the model chosen for enhancing each frame entails an overall improved performance. In [11], they employ multiple noise-specific regression models and use a DNN-based classifier to find the model matching closest to the input noise for robust SNR estimation. They use this classifier to get the closest matching noise model in the case of an unknown noise. But this technique does not ameliorate the original problem of mismatch between the training and testing conditions.

Nazreen and Ramakrishnan [12] report on their preliminary experiments where, Mote Carlo dropout proposed by Gal and Ghahramani [13] is used for modeling uncertainty in each noise-specific DNN model. Monte Carlo dropout, unlike conventional dropout [14], [15] uses dropout during inference stage and multiple forward passes of the input are carried out dropping random neurons of the network each time. The output samples could be considered as Monte Carlo (MC) samples from the model posterior [16]. In [12], a measure of the model uncertainty obtained from the output samples of each noise-specific DNN model is used to pick the appropriate model for enhancing a noisy speech frame (Var-MC). For the uncertainty measurement, the trace of the covariance matrix of the output samples (Var) is used [16].

Even though the above Var-MC algorithm gives superior performance than a DNN classifier-based selection scheme for unseen noise cases, the algorithm gives poorer performance for seen noise cases than the classifier-based scheme. In order to rectify this, we propose a conditional selection criterion for the noise models in which the selection of noise models can be switched from model uncertainty-based to classifier-based. We show our results on unseen as well as seen noises. We also show some promising results in the case where speech is corrupted by a mixture of noises and a non-stationary scenario where random segments of speech are affected by different unseen noises. In another real world experiment, we record real world, traffic noise and add to clean speech in order to test our models.

II. μ-MC: A Var threshold (μ) based algorithm to choose either classifier-based or model-uncertainty-based selection of model

A threshold is set on the Var values of all the five models and based on this, one could go for either a classifier-based selection or Var based selection, as shown in Fig. 1.

The magnitude STFT of the input noisy frame $Y_f \in \mathbb{R}^{K\times 1}$, is passed through all the M available MC dropout models (five for our experiments) J different times, by dropping out random units each time. The corresponding outputs are $\{\hat{S}_j^i(Y_f)\}; 1 \leq j \leq J; 1 \leq i \leq M$; where i is the model index and $M = 5$. If the $Var(S^i)$ values of all the five models are above a threshold μ, this could be an indication that the noise corrupting that frame does not match with any of these M models and hence it is an unseen noise. In such a case, the model which gives the minimum Var value is considered as the best model for enhancing that frame. The enhanced output

is obtained by taking the empirical mean of the J outputs of the corresponding model: $\{\hat{S}_j^{i^*}(Y_f)\}; 1 \leq j \leq J; 1 \leq i^* \leq M$.

On the other hand, if the Var values are below the threshold μ, it is possible that the input noisy frame is corrupted by a noise matching one of these models (seen noise) and hence we could go for the classifier-based selection, as Var-MC performance is not reliable on seen noises. The input noisy frame Y_f is first fed into the classifier which picks the best model c^* for enhancing the frame. Let the outputs of the corresponding model be; $\{\hat{S}_j^{c^*}(Y_f)\}; 1 \leq j \leq J; 1 \leq c^* \leq M$. The enhanced frame $\hat{S}_C(Y_f)$ is obtained by taking the empirical mean of these J different outputs. Inverse Fourier transform is applied on $\hat{S}$ with the noisy phase information to obtain the enhanced output.

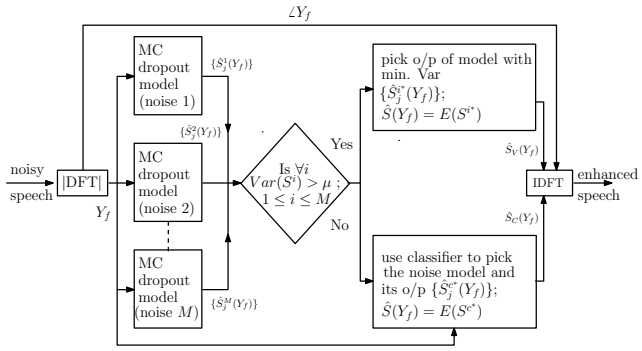

Fig. 1. μ-MC : A Var threshold (μ) based algorithm for enhancement using multiple models trained on distinct noises. The appropriate model output is selected for each input frame of noisy speech, using model uncertainty as a selection criterion, or a noise classifier.

III. Details of the experiments conducted

For our experiments, we use TIMIT [17] speech corpus which consists of 6300 sentences from 630 speakers with the train and test sets containing 4620 and 1680 utterances, respectively. The entire TIMIT training data is used for training and 50 test files are randomly chosen from the TIMIT test utterances for testing. To add noise to the speech we use NOISEX-92 [18] database. For our experiments, an additive noisy framework is assumed. The noise files are downsampled to 16 kHz so as to match the sampling rate of TIMIT, in-order to synthesize noisy test and training speech data. We use magnitude STFT to train each DNN model computed using a frame size of 30 ms with 10 ms frame shift after applying a Hamming window. A 512-point FFT is used and the first 257 points are used to train each DNN model due to the symmetry of the spectrum.

During the inference stage, the number of repetitions for MC dropout models J, is chosen as 50. Each DNN based regression model is trained with the magnitude STFT of noisy speech as input and clean speech as target. The Adam optimizer [19] is chosen. The dropout rate is set to 20%.

The testing is done using TIMIT test set corrupted with unseen noises white, pink and factory1 and seen noises factory2, m109 and leopard (babble and volvo noise results are omitted due to space constraints) at SNRs varying from -10dB to 10 dB. We also evaluate the algorithm on a real world, traffic noise which we have recorded. The results are reported in terms of segmental SNR (SSNR) [20].

A. DNN architecture for enhancement

Each DNN model for enhancement consists of 3 fully connected layers of 2048 neurons and an output layer of 257. ReLu activation function is used in all the three layers as well as the output layer due to the nonnegative nature of magnitude STFT. We minimize the mean square logarithmic error (E_l) loss function between the noisy and clean magnitude spectra. The architecture is based on the best performing DNN configuration in [6].

B. Classifier-based model selection for comparison

We compare the $\mu - MC$ and Var-MC results to that of the case where a DNN classifier is used to pick the noise-specific models which can either be trained on conventional dropout (class-C) or MC dropout (class-MC). The DNN classifier consists of 3 fully connected layers of 2048 neurons and an output layer of 5 neurons for the five noises. ReLu activation function is used in all the three layers and Softmax activation function is used in the output layer. Categorical cross entropy is used as the loss function. The classifier is trained on speech corrupted with factory2, babble, leopard, m109 and volvo noises at SNRs 0, 5 and 10 dB which we consider as the seen noises.

C. Var-MC and μ-MC experimental setup

For Var-MC and μ-MC experiments, we train five different DNN models separately on speech corrupted with factory2, m109, leopard, babble and volvo noises, each at SNRs 0, 5 and 10 dB. Each of these DNN model is trained using MC dropout as well as conventional dropout [14], [15], for comparison using the entire TIMIT training data. The architecture of the models are as defined in section III-A.

The threshold μ is selected based on the experiments on a validation set of 30 files from TIMIT corrupted with seen noises factory 2, m109, leopard, babble and volvo and unseen pink noise at SNRs -10, -5 , 0, 5 and 10 dB. For our experiments, this threshold is set at $\mu = 0.16$.

IV. Results and discussion

A. Results of μ-MC model on unseen and seen noises

Table I shows the performance of μ-MC compared to class-C, class-MC and Var-MC in terms of SSNR [20] for unseen noises white, pink and factory1. The results are averaged over 50 files randomly selected from TIMIT [17] test set. Var-MC gives the best performance of all, especially at lower SNRs. But at higher SNRs like 5 and 10 dB for example, the performance of Var-MC drops below those of class-MC and class-C in some cases. μ-MC algorithm not only compensates for this performance drop, but also gives performance superior to class-C and class-MC at lower SNRs even though it is not as much as that of Var-MC at lower SNRs.

TABLE I

RESULTS ON UNSEEN AND SEEN NOISES: PERFORMANCE COMPARISON (IN TERMS OF **SSNR: SEGMENTAL SNR**) OF VAR-MC AND μ-MC ALGORITHMS WITH CLASS-C AND CLASS-MC FOR SPEECH CORRUPTED WITH UNSEEN NOISES WHITE, PINK AND FACTORY1 AS WELL AS SEEN NOISES FACTORY2, LEOPARD AND M109, AT SNRS -10, -5, 0, 5 AND 10 *dB* AVERAGED OVER 50 FILES RANDOMLY SELECTED FROM TIMIT TEST SET.

	White (Unseen)					Pink (Unseen)					Factory1 (Unseen)				
SNR (dB)	**Noisy input**	**Class-C**	**Class-MC**	**Var-MC**	**μ-MC $\mu = 0.16$**	**Noisy input**	**Class-C**	**Class-MC**	**Var-MC**	**μ-MC $\mu = 0.16$**	Noisy input	**Class-C**	**Class-MC**	**Var-MC**	**μ-MC $\mu = 0.16$**
10	2.0	2.6	2.6	**2.7**	**2.7**	2.2	4.8	4.8	4.5	4.7	2.3	4.9	4.9	4.8	**4.9**
5	-1.6	-0.8	-0.8	**-0.7**	**-0.7**	-1.4	1.7	1.7	1.6	**1.7**	-1.3	2.0	2.0	**2.0**	**2.0**
0	-4.6	-4.1	-4.0	**-3.8**	**-4.0**	-4.5	-1.6	-1.6	**-1.3**	**-1.6**	-4.4	-1.1	-1.1	**-0.83**	**-1.1**
-5	-7.2	-6.7	-6.6	**-6.5**	**-6.6**	-7.1	-4.5	-4.5	**-3.7**	**-4.5**	-6.9	-4.1	-4.1	**-3.3**	**-4.0**
-10	-8.9	-8.7	-8.6	**-8.4**	**-8.5**	-8.8	-7.1	-7.1	**-5.4**	**-6.9**	-8.7	-6.6	-6.6	**-5.3**	**-6.3**
	Factory 2 (Seen)					**Leopard (Seen)**					**M109 (Seen)**				
SNR (dB)	**Noisy input**	**Class-C**	**Class-MC**	**Var-MC**	**μ-MC $\mu = 0.16$**	**Noisy input**	**Class-C**	**Class-MC**	**Var-MC**	**μ-MC $\mu = 0.16$**	**Noisy input**	**Class-C**	**Class-MC**	**Var-MC**	**μ-MC $\mu = 0.16$**
10	2.6	9.5	9.5	8.1	**9.5**	2.5	8.9	8.9	8.5	**8.9**	2.5	9.1	9.1	8.1	**9.1**
5	-0.9	7.7	7.7	5.8	7.6	-1.1	7.4	7.4	7.0	**7.4**	-1.1	7.3	7.3	6.3	**7.3**
0	-4.1	5.8	5.8	3.3	**5.8**	-4.3	5.9	5.9	5.6	**5.9**	-4.2	5.3	5.3	4.3	**5.3**
-5	-6.7	4.0	4.0	1.3	3.9	-6.8	4.3	4.4	4.2	**4.3**	-6.8	3.5	3.5	2.5	**3.5**
-10	-8.5	2.1	2.1	0.5	**2.1**	-8.6	2.7	2.9	2.7	**2.8**	-8..6	1.9	1.9	1.0	**1.9**

TABLE II

MIXED, NON-STATIONARY UNSEEN AND REAL WORLD TRAFFIC NOISE EXPERIMENTS: PERFORMANCE EVALUATION (IN TERMS OF **SSNR: SEGMENTAL SNR**) OF VAR-MC AND μ-MC ALGORITHMS FOR THE CASES; MIX: SPEECH IS CORRUPTED WITH A MIXTURE OF UNSEEN NOISES, FACTORY1 AND PINK; TV: EACH TEST UTTERANCE OF DURATION 2 TO 3 SEC. IS DIVIDED INTO A RANDOM NUMBER (5 TO 10) OF SEGMENTS OF RANDOM LENGTH AND UNSEEN NOISES WHITE, FACTORY1 AND PINK ARE ADDED RANDOMLY TO THESE SEGMENTS; TRAFFIC: REAL WORLD TRAFFIC NOISE IS RECORDED AND CLEAN SPEECH IS CORRUPTED WITH THIS NOISE. THE RESULTS AVERAGED OVER 50 FILES RANDOMLY SELECTED FROM TIMIT TEST SET.

	Mix: Additive noise Factory1+Pink (unseen)					TV: White-Factory1-Pink (unseen)					Traffic (unseen)				
SNR (dB)	**Noisy input**	**Class-C**	**Class-MC**	**Var-MC**	**μ-MC $\mu = 0.16$**	**Noisy input**	**Class-C**	**Class-MC**	**Var-MC**	**μ-MC $\mu = 0.16$**	**Noisy input**	**Class-C**	**Class-MC**	**Var-MC**	**μ-MC $\mu = 0.16$**
10	2.2	4.8	4.8	4.6	**4.8**	3.0	4.9	4.9	4.7	**4.9**	3.4	4.9	4.9	**5.0**	**5.0**
5	-1.3	1.8	1.8	**1.8**	**1.8**	-0.6	1.9	1.9	**1.9**	**1.9**	-0.2	2.0	2.0	**2.2**	**2.0**
0	-4.5	-1.3	-1.3	**-1.0**	**-1.3**	-3.9	-1.4	-1.4	**-1.2**	**-1.4**	-3.4	-1.1	-1.1	**-0.8**	**-1.0**
-5	-7.0	-4.3	-4.3	**-3.5**	**-4.1**	-6.5	-4.3	-4.3	**-3.7**	**-4.2**	-6.0	-4.1	-4.1	**-3.6**	**-3.9**
-10	-8.8	-6.8	-6.8	**-5.5**	**-6.5**	-8.4	-6.9	-6.9	-5.6	**-6.7**	-7.9	-6.6	-6.6	**-6.1**	**-6.2**

Table I also shows the performance of Var-MC and μ-MC for seen noises factory2, m109 and leopard (babble and volvo noise results are omitted due to space constraints). It can be seen that Var-MC performance is really poor compared to class-C and class-MC for seen noises. μ-MC algorithm compensates for this performance drop by using per frame threshold μ to select between the Var-MC and class-MC schemes. Thus μ-MC algorithm not only gives performance superior to class-C and class-MC algorithm for unseen cases but also gives comparable performance for seen noise cases.

B. Results of Var-MC and μ-MC on mixed, time varying and real world, traffic noises

We have also carried out some experiments to evaluate the performance of both Var-MC and μ-MC algorithms where a mixture of factory 1 and pink noise (mix) affects the speech. In another experiment, we show the evaluation on a non-stationary scenario, where each test utterance of 2 to 3 seconds is divided into a random number (chosen to lie between 5 and 10) of segments of random lengths. One among the unseen noises factory1, pink or white is randomly chosen to affect these segments (TV). Table II shows these results. In both the cases mix and TV, we see that both Var-MC and μ-MC give performance superior to class-C and class-MC. Though Var-MC gives the best performance of all at lower SNRs, at higher SNRs this performance drops below those of class-C and class-MC. μ-MC compensates for this performance drop and gives performance comparable to class-C and class-MC at higher SNRs.

We also have performed a real world experiment, where we record real world, traffic noise to evaluate the performance of our algorithm. We believe this experiment is significant, since in most cases DNNs for speech enhancement might be untrained on these real world noises. Results reported by Table II show that both Var-MC and μ-MC give performances superior to class-MC and class-C at all SNRs.

V. CONCLUSION

We propose a Var threshold μ based algorithm (μ-MC) to switch between model uncertainty-based and classifier-based selection scheme, in order to compensate for the poor performance of the model uncertainty-based DNN model selection scheme (Var-MC) proposed in [12], compared to classifier-based selection scheme, for enhancement of speech corrupted with seen noises. We show the performance of μ-MC algorithm for unseen and seen noises at SNRs varying from -10 dB to 10 dB. We also show some promising results for the cases where speech is corrupted by a mixture of unseen noises and where random segments of speech are corrupted by random unseen noises. In another significant result, we show the performance of the algorithms on speech corrupted by real world, traffic noise.

REFERENCES

[1] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.

[2] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural computation*, vol. 18, no. 7, pp. 1527–1554, 2006.

[3] A. L. Maas, Q. V. Le, T. M. O'Neil, O. Vinyals, P. Nguyen, and A. Y. Ng, "Recurrent neural networks for noise reduction in robust ASR," in *13th Annual Conf. , International Speech Communication Association*, 2012.

[4] Y. Wang and D. Wang, "Towards scaling up classification-based speech separation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 7, pp. 1381–1390, 2013.

[5] Y. Xu, J. Du, L.-R. Dai, and C.-H. Lee, "An experimental study on speech enhancement based on deep neural networks," *IEEE Signal processing letters*, vol. 21, no. 1, pp. 65–68, 2014.

[6] Y. Xu, J. Du, L.-R. Dai, and C.-H. Lee, "A regression approach to speech enhancement based on deep neural networks," *IEEE/ACM Trans. Audio, Speech and Language Processing (TASLP)*, vol. 23, no. 1, pp. 7–19, 2015.

[7] P. M. Nazreen, A. G. Ramakrishnan, and P. K. Ghosh, "A class-specific speech enhancement for phoneme recognition: A dictionary learning approach," *Proc. Interspeech*, pp. 3728–3732, 2016.

[8] P. M. Nazreen, A. G. Ramakrishnan, and P. K. Ghosh, "A joint enhancement-decoding formulation for noise robust phoneme recognition," *14th IEEE India Council Inter. Conf. (INDICON)*, 2017.

[9] Z.-Q. Wang, Y. Zhao, and D. Wang, "Phoneme-specific speech separation," in *IEEE Inter. Conf. Acoustics, Speech and Signal Processing*, 2016.

[10] S. E. Chazan, S. Gannot, and J. Goldberger, "A phoneme-based pre-training approach for deep neural network with application to speech enhancement," in *2016 IEEE International Workshop on Acoustic Signal Enhancement (IWAENC)*. IEEE, 2016, pp. 1–5.

[11] P. Papadopoulos, A. Tsiartas, and S. Narayanan, "Long-term SNR estimation of speech signals in known and unknown channel conditions," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 12, pp. 2495–2506, 2016.

[12] P. Nazreen and A. Ramakrishnan, "DNN based speech enhancement for unseen noises using monte carlo dropout," in *2018 12th International Conference on Signal Processing and Communication Systems (ICSPCS)*. IEEE, 2018, pp. 1–6.

[13] Y. Gal and Z. Ghahramani, "Dropout as a Bayesian approximation: Representing model uncertainty in deep learning," in *International Conference on Machine Learning*, 2016, pp. 1050–1059.

[14] G. E. Dahl, T. N. Sainath, and G. E. Hinton, "Improving deep neural networks for LVCSR using rectified linear units and dropout," in *Acoustics, Speech and Signal Processing (ICASSP), International Conference on*. IEEE, 2013, pp. 8609–8613.

[15] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[16] A. Kendall and R. Cipolla, "Modelling uncertainty in deep learning for camera relocalization," in *Robotics and Automation (ICRA), International Conference on*. IEEE, 2016, pp. 4762–4769.

[17] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, and D. S. Pallett, "DARPA TIMIT acoustic-phonetic continous speech corpus CD-ROM. NIST speech disc 1-1.1," *NASA STI/Recon Technical Report N*, vol. 93, p. 27403, Feb. 1993.

[18] A. Varga and H. J. M. Steeneken, "Assessment for automatic speech recognition ii: Noisex-92: A database and an experiment to study the effect of additive noise on speech recognition systems," *Speech Commun.*, vol. 12, no. 3, pp. 247–251, Jul. 1993. [Online]. Available: http://dx.doi.org/10.1016/0167-6393(93)90095-3

[19] D. P. Kingma and J. L. Ba, "Adam: A method for stochastic optimization," *Proc. the ICLR 2015*, pp. 1–13, 2015.

[20] Y. Hu and P. C. Loizou, "Evaluation of objective quality measures for speech enhancement," *IEEE Transactions on audio, speech, and language processing*, vol. 16, no. 1, pp. 229–238, 2008.