INTEGRATED DNN-BASED MODEL ADAPTATION TECHNIQUE FOR NOISE-ROBUST SPEECH RECOGNITION

Kang Hyun Lee, Woo Hyun Kang, Tae Gyoon Kang and Nam Soo Kim

Department of Electrical and Computer Engineering and INMC,
Seoul National University, 1 Gwanak-ro, Gwanak-gu, Seoul, Korea, 08826
E-mail: {khlee, whkang, tgkang}@hi.snu.ac.kr, nkim@snu.ac.kr

ABSTRACT

Since the introduction of deep neural network (DNN)-based acoustic model, robust automatic speech recognition using DNN are being in research. Especially in model adaptation, the techniques utilizing auxiliary context features is known to be a promising technique. Recently, we proposed a technique which is called two-stage noise-aware training (TS-NAT). The key idea of TS-NAT is to let the DNN clarify the relationship among noise estimate, noisy features and phonetic target through clean feature representation. However, although TS-NAT enhances the robustness of the DNN, we cannot be certain whether TS-NAT describes the clean feature representation sufficiently. In this paper, we extend TS-NAT using true noise feature and various DNN training techniques. It has been shown that the proposed technique outperforms the conventional DNN-based techniques on Aurora5-task and mismatched noise conditions.

Index Terms— Deep neural networks (DNNs), robust speech recognition, noise-aware training (NAT), multi-task learning (MTL), joint training.

1. INTRODUCTION

Ever since the deep neural network (DNN)-based acoustic model appeared, the recognition performance of automatic speech recognition (ASR) has been greatly improved [1, 2, 3, 4]. Based on this achievement, researches on DNN-based techniques for noise robustness are also in progress. Among various approaches, adaptation technique employing auxiliary features with acoustic context information demonstrated their potential [5, 6, 7, 8, 9, 10, 11, 12].

One of the simplest methods of these approaches is to augment the auxiliary features with the input vector of the network [5, 6]. As an example, the technique referred to as noise-aware training (NAT) attained state-of-the-art results on Aurora-4 task [5]. NAT enables the DNN to learn the relationship among noisy input, noise features and target vectors corresponding to the phonetic identity by augmenting an estimate of the noise present in the input signal. Due to its simple implementation and good performance, NAT has already been applied actively in speech enhancement and robust ASR [13].

Also, target vectors of DNN can augment the auxiliary [14]. In other words, the network is trained to solve multiple tasks simultaneously using a shared set of parameters [7, 8]. This technique is called multi-task learning (MTL). By learning multiple tasks in parallel, the DNN parameters can learn additional information about the domain using the target vectors of the secondary task. Meanwhile, the performance of DNN can be enhanced by giving an appropriate intermediate concept which the DNN should represent in the mid-level [15]. Due to this reason, recent researches on joint training technique of DNN have drawn attention [9, 10]. The joint training technique builds a DNN by concatenating two independently trained DNNs and jointly adjusting the parameters.

However, simply setting the auxiliary features as the input to a DNN may not necessarily be the best approach for exploiting context information. In this sense, we have recently proposed a novel approach which takes advantage of the inherent robustness of the NAT framework more efficiently [16], called the two-stage noise-aware training (TS-NAT). The key idea of TS-NAT is to let the DNN clarify the relationship among noise estimate, noisy features and phonetic targets through the clean feature representation. In order to accomplish this, TS-NAT cascades two individually fine-tuned DNNs into a single DNN. The first DNN performs reconstruction of the clean features from noisy features when noise estimates are augmented, and the next DNN attempts to learn the mapping between the reconstructed features and the phonetic targets. It has been shown that TS-NAT outperforms the conventional NAT on Aurora-5 task.

It is certain that the noise estimate feature contributes to the clean feature reconstruction. However, we cannot deny that there exists a limitation in describing the clean feature representation from the information of noisy and noise estimate features. This distorts the reconstruction of clean feature and consequently interrupts the connection between input of the DNN and the corresponding phonetic target. Particularly,

This work was supported in part by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MEST) (NRF-2015R1A2A1A15054343).

this can be a serious problem in mismatched noise conditions.

In this paper, we extend our previous work for supplementing the aforementioned issue of TS-NAT. The main idea of the proposed approach is to design a delicate DNN-based acoustic model which is aware of real noise information and robust to mismatched noise conditions. To implement this, the proposed approach combines TS-NAT with two well-known DNN-based adaptation techniques, MTL and joint training [7, 8, 9, 10]. Also, we used rectified linear unit (ReLU) which is known to have better performance than sigmoid in a very deep network[17].

The performance of the proposed approach is evaluated on the Aurora-5 task and mismatched noise conditions, and better performance was observed compared to the conventional DNN techniques.

2. RELATED RESEARCH

In this work, for a simple problem formulation, we will only consider acoustic environments where the background noises are dominant factors of speech degradation. Let $\mathbf{y}_t$, $\mathbf{x}_t$, $\mathbf{n}_t$ and $\mathbf{s}_t$ denote the observed noisy feature, the corresponding unknown clean feature, the corrupting noise and the HMM state identity being extracted at the t-th frame , respectively. Additionally, we denote a subsequence of vectors $\mathbf{x}_{m_1}\mathbf{x}_{m_1+1}\cdots\mathbf{x}_{m_2}$ from frame index m_1 to m_2 as $\mathbf{x}_{m_1}^{m_2}$.

2.1. Noise-aware training

Under the general framework of HMM-based recognition, we assume that there exists an unknown underlying function that approximates the posterior probabilities of the HMM states given as follows:

$$p(\mathbf{s}_t|\mathbf{y}_1^T) \cong f(\mathbf{y}_{t-\tau}^{t+\tau}, \mathbf{n}_{t-\tau}^{t+\tau}) \tag{1}$$

where $f(\cdot)$ represents the function that maps the noisy and noise features to the corresponding HMM state identity which contains phonetic information, T denotes the length of the input feature, and the subscript τ represents the temporal coverage which is required for figuring out the contextual information of the speech signal.

Since the true noise features $\mathbf{n}_{t-\tau}^{t+\tau}$ in (1) are unknown, NAT replaces them with a single noise estimate. The input vector of NAT is formed by augmenting the noise estimate with a window of consecutive frames of noisy feature, i.e.,

$$\mathbf{v}_t = [\mathbf{y}_{t-\tau}^{t+\tau}, \widehat{\mathbf{n}}_t] \tag{2}$$

where $\mathbf{y}_{t-\tau}^{t+\tau}$ represents a window of 2τ + 1 frames of noisy speech features and $\widehat{\mathbf{n}}_t$ represents the noise estimate.

2.2. Two-stage noise-aware training

The basic idea of TS-NAT starts from the assumption that the underlying function $f(\cdot)$ in (1) can be expressed as a composition of two separate functions as follows:

$$p(\mathbf{s}_t|\mathbf{y}_1^T) \cong f(\mathbf{y}_{t-\tau}^{t+\tau}, \mathbf{n}_{t-\tau}^{t+\tau}) \cong h \circ g(\mathbf{y}_{t-\tau}^{t+\tau}, \mathbf{n}_{t-\tau}^{t+\tau}) \tag{3}$$

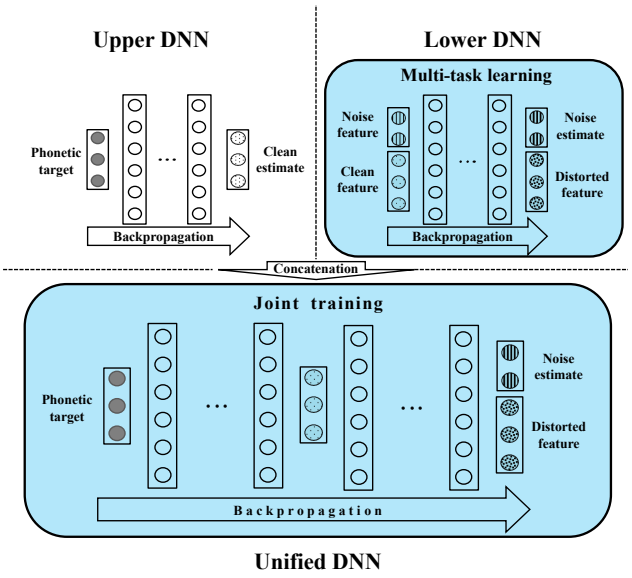

Fig. 1. DNN structure of TS-NAT and its extension.

where the output of $g(\cdot)$ is a clean feature vector stream,

$$\mathbf{x}_{t-\tau}^{t+\tau} \cong g(\mathbf{y}_{t-\tau}^{t+\tau}, \mathbf{n}_{t-\tau}^{t+\tau}), \tag{4}$$

and

$$p(\mathbf{s}_t|\mathbf{y}_1^T) \cong h(\mathbf{x}_{t-\tau}^{t+\tau}). \tag{5}$$

In (3)-(5), $g(\cdot)$ represents a function maps the noisy and noise features to the clean speech features and $h(\cdot)$ is a function that predicts the phonetic target based on the clean speech feature stream.

The unified DNN is constructed by cascading two individually fine-tuned DNNs where each DNN approximates the function $g(\cdot)$ and $h(\cdot)$ in (3). The first DNN which is called the lower DNN, separates the clean speech features from the corruption noises. For training the lower DNN, we apply the deep denoising autoencoder which has proven its capability of reducing the distortion in the original noisy feature [18]. The second DNN which is called the upper DNN, models the relationship between the output vector generated by the lower DNN and the phonetic target.

3. PROPOSED APPROACH

Although TS-NAT enhances the robustness of the DNN, we cannot be certain whether TS-NAT describes the clean feature representation sufficiently. Unlike expression (4), we replaced the true noise feature stream with noise estimate features in the actual implementation. Therefore, actual mapping from noisy and noise estimate features to clean features can be expressed as follows:

$$l(\mathbf{y}_{t-\tau}^{t+\tau}, \widehat{\mathbf{n}}_t) = \widehat{\mathbf{x}}_{t-\tau}^{t+\tau} = \mathbf{x}_{t-\tau}^{t+\tau} + v_{t-\tau}^{t+\tau} \tag{6}$$

where $l(\cdot)$ represents the actual function of the lower DNN that reconstructs clean features from the noisy and noise estimate features and υ is the reconstruction error between the clean and the clean estimate features. In other words, insufficient information about the true noise makes the lower DNN distort reconstructed clean features and this naturally leads to improper mapping between the input and phonetic target. To compensate for this problem, we extend TS-NAT by applying the techniques described in the section below. In this work, some individual modifications are applied to both the lower and upper networks to improve the whole system. Therefore, we call these two networks as extended lower and upper network respectively.

3.1. Multi-task learning

In a general MTL framework, multi-task objective function J_{MTL} is expressed as follows:

$$J_{MTL} = J + \alpha J_{aux} \tag{7}$$

where J and J_{aux} denote the objective functions of primary and secondary tasks respectively, and α is the weight parameter which determines how much importance the secondary task has. After the training is over, only the primary task is performed and the parameters associated with the output of the secondary task are discarded.

In this work, MTL is applied to the lower DNN with true noise feature. Specifically, the target vector of the lower DNN adds noise feature corresponding to noise estimate feature of the input vector. Therefore, the objective function of the extended lower DNN J_L can be represented as follows:

$$J_L = \sum_t ||\mathbf{o}_t - \widehat{\mathbf{o}}_t||^2 + \alpha \sum_t ||\mathbf{n}_t - \widehat{\mathbf{n}}_t||^2 \tag{8}$$

where $\mathbf{o}_t$ and $\widehat{\mathbf{o}}_t$ denote the target and output vectors of the lower DNN. By flowing back the information of the true noise feature, the extended lower DNN can absorb the environmental information more distinctly. Particularly, the shared structure serves to improve the generalization of the model and its accuracy on an unseen test set [7]. In this technique, α was set to 1.

3.2. Joint training

As in TS-NAT, the extended upper DNN learns the mapping between the output vector of the extended lower DNN $\widehat{\mathbf{o}}_t$ and the corresponding one-hot encoding label which contains information of the HMM states.

After training the extended upper DNN, two different networks are cascaded to form a single larger DNN and the unified DNN jointly adjusts the weights using the backpropagation algorithm. In detail, the error signal between the phonetic target and the output of the unified DNN flows back to the clean estimate feature layer and the extended lower DNN, consequently training all the parameters. With this series of processes, learning the relationship among the noisy, noise estimate, true noise features and phonetic target labels can be enhanced by guiding the DNN through the intermediate level features [15].

4. EXPERIMENTS

To evaluate the speech recognition performance of the proposed approach, we performed a series of experiments in Aurora-5 task [19].

4.1. Aurora-5 task and GMM-HMM system

The Aurora-5 task was developed to investigate the performance of speech recognition for speech recorded with hands-free devices in noisy room environments. The test data consisted of two sets: G. 712 filtered and non-filtered sets. The G. 712 filtered set comprised of clean speech utterances where randomly selected car or public space noise samples were added at signal-to-noise ratio (SNR) levels of 0 to 15 dB. The non-filtered set consisted of clean speech utterances where randomly selected interior noises were augmented at the same SNR range mentioned above.

In these experiments, we used multi-condition training data for construction of all the DNN-based acoustic models. In order to create phonetic labels of the training data, the GMM-HMM systems were built based on the clean speech data provided by the G. 712 filtered and non-filtered data sets which is counterpart of multi-condition training data. These systems consisted of 179 HMMs states and 4 Gaussians per state trained using maximum likelihood estimation. The number of utterances used for HMM training was 8623 for each data set. The input features were 39-dimensional MFCC features (static plus first and second order delta features) and cepstral mean normalization was performed. The training of the HMM parameters and Viterbi decoding for speech recognition was carried out using HTK [20].

4.2. Training and structures of DNN-based techniques

The performance of the proposed method was compared with three different versions of DNN-based approaches. The compared techniques are

- *NAT*: Noise-aware training [5],
- *NAT + MTL*: Combination of NAT and MTL to perform both the primary acoustic modeling task and the secondary feature enhancement task,
- *TS-NAT*: Two-stage noise-aware training [16],

For training all the DNN-based acoustic models, log mel filterbank (FBANK) feature of 23-dimension was used. As in the case of MFCC feature above, both the first and second-order derivative of FBANK features were used. The input layers for all the techniques had a total of 828 visible units

Table 1. WERs (%) on Aurora-5 task according to variety of DNN-based acoustic models

SNR (dB)	Non-filtered								G.712 filtered							
Method	*NAT*		*NAT + MTL*		*TS-NAT*		*Proposed*		*NAT*		*NAT + MTL*		*TS-NAT*		*Proposed*	
Dropout	0 %	20 %	0 %	20 %	0 %	20 %	0 %	20 %	0 %	20 %	0 %	20 %	0 %	20 %	0 %	20 %
Clean	1.42	1.28	1.23	1.12	0.93	0.95	0.95	**0.89**	0.92	0.78	0.82	0.71	0.72	0.71	**0.65**	0.70
15	1.88	1.87	1.71	1.73	1.54	1.50	1.49	**1.28**	1.25	1.18	1.11	1.10	0.95	0.90	0.89	**0.82**
10	3.34	3.14	3.10	2.94	2.90	2.58	2.74	**2.35**	2.15	1.87	1.98	1.76	1.56	**1.28**	1.52	1.37
5	7.87	7.55	7.42	7.28	7.12	6.60	6.88	**6.23**	4.52	4.35	4.31	4.21	4.03	3.65	3.89	**3.52**
0	20.73	20.01	20.21	19.78	19.62	19.08	19.14	**18.87**	12.67	12.25	12.24	12.02	11.90	11.54	11.58	**11.29**
Average	7.05	6.77	6.73	6.57	6.42	6.14	6.24	**5.92**	4.30	4.09	4.09	3.96	3.83	3.67	3.71	**3.54**

Table 2. WERs (%) on the noise-mismatched test set according to variety of DNN-based acoustic models

SNR (dB)	Non-filtered								G.712 filtered							
Method	*NAT*		*NAT + MTL*		*TS-NAT*		*Proposed*		*NAT*		*NAT + MTL*		*TS-NAT*		*Proposed*	
Dropout	0 %	20 %	0 %	20 %	0 %	20 %	0 %	20 %	0 %	20 %	0 %	20 %	0 %	20 %	0 %	20 %
Clean	1.42	1.28	1.23	1.12	0.93	0.95	0.95	**0.89**	0.92	0.78	0.82	0.71	0.72	0.71	**0.65**	0.70
15	3.18	3.12	2.95	2.92	2.87	2.80	2.81	**2.62**	4.44	4.29	4.05	3.98	4.18	**3.97**	4.30	4.05
10	5.98	5.88	5.65	5.64	5.55	5.45	5.51	**5.01**	10.78	10.35	10.28	9.78	8.83	8.23	8.12	**7.89**
5	12.43	12.11	12.21	11.85	11.72	11.36	10.98	**10.56**	18.82	18.12	18.04	17.78	17.12	15.85	15.23	**14.89**
0	25.24	24.02	24.76	23.75	22.26	21.50	21.24	**19.76**	30.23	29.75	29.88	29.25	29.11	28.25	27.57	**26.78**
Average	9.65	9.28	9.36	9.06	8.67	8.41	8.30	**7.77**	13.04	12.66	12.61	12.30	11.99	11.40	11.17	**10.86**

by augmenting a context window of 11 consecutive FBANK features with the IMM-based noise estimate [21].

Among these techniques, *NAT + MTL* is a technique which uses parallel training data for MTL, where the network is trained to perform both the primary acoustic modeling task and the secondary feature enhancement task with the input vector (2). This can be interpreted as a conventional model adaptation approach which utilizes same resources with our proposed technique [8]. The target vector dimension of *NAT + MTL* is 248 by adding the corresponding clean feature of current frame (69 dim.) to the phonetic target (179 dim.).

All the techniques had 11 hidden layers with 2048 ReLUs except for the intermediate layers of 759 linear units from *TS-NAT* and the proposed technique. The final output layers of the techniques had soft-max 179 units, each corresponding to the state of the HMM systems. The parameters of the DNN-based techniques were randomly initialized and fine-tuned using stochastic gradient descent (SGD) algorithm.

Mini-batch size for the SGD algorithm was set to be 256 for all of the DNN-based techniques. The momentum was set to be 0.5 at the first epoch and increased to 0.9 afterward. The learning rate was initially set to be 0.01 and exponentially decayed over each epoch with a decaying factor of 0.9 except for the cases of two lower DNNs and joint training of the proposed method. For two lower DNNs and the joint training, learning rate was initially set to be 0.0005 and exponentially decayed over each epoch with a decaying factor of 0.95. All the training of DNN-based techniques were stopped after 50 epochs.

All the techniques evaluated in this experiments were based on wide and very deep DNN structures. To prevent overfitting, dropout was also applied [22], [23]. The retention rate of dropout was 0.8.

4.3. Evaluations

Table 1 shows the results of the various DNN-based techniques. We can see that the proposed method outperformed other DNN-based techniques irrespective of the SNRs. Further improvement was observed when the dropout training was applied. With dropout training performed, the average relative error rate reductions (RERRs) of *Proposed* over *NAT + MTL* were 9.83% and 10.61% in non-filtered and G.712 filtered set.

To evaluate the proposed technique in training-test mismatched noise conditions, we constructed the noise-mismatched test sets by mixing the clean speech of non-filtered and G. 712 filtered sets with four noises included in 100 non-speech environmental sounds [24]. Four types of noise were chosen from 100 noise types : animal, water, wind sound and phone dialing. Each noise types were added to the G. 712 filtered and non-filtered sets at SNRs between 0 and 15 dB with equal rate. From the results in Table 2, we can see that the proposed technique is more effective in mismatched noise conditions. Especially, when dropout training is performed the average relative error rate reductions (RERRs) of *Proposed* over *NAT + MTL* were 14.22% and 11.69% in noise-mismatched non-filtered and G.712 filtered set.

5. CONCLUSION

In this paper, we proposed an extension of TS-NAT which supplements the information needed for clean feature representation. Through a series of experiments, we have found that the proposed technique outperforms the conventional techniques in word accuracy on the Aurora-5 task and mismatched noise conditions. Future study will deal with techniques considering other environmental factors such as reverberation.

6. REFERENCES

[1] A. Mohamed, G. E. Dahl, and G. E. Hinton, "Acoustic modeling using deep beliefs networks," *IEEE Trans. Audio, Speech, Language Process.*, vol. 20, no. 1, pp. 14-22, Jan. 2012.

[2] G. E. Dahl, D. Yu, L. Deng, and A. Acero, "Context-dependent pre-trained deep neural networks for large vocabulary speech recognition," *IEEE Trans. Audio, Speech, Language Process.*, vol. 20, no. 1, pp. 30-42, Jan. 2012.

[3] B. Kingsbury, T. N. Sainath, and H. Soltau, "Scalable minimum Bayes risk training of deep neural network acoustic models using distributed Hessian-free optimization," in *Proc. Interspeech*, 2012, pp. 10-13.

[4] D. Yu, M. L. Seltzer, J. Li, J. Huang, and F. Seide, "Feature learning in deep neural networks - A study on speech recognition tasks," *CORR*, vol. abs/1301.3605, 2013.

[5] M. Seltzer, D. Yu, and Y. Wang, "An investigation of deep neural networks for noise robust speech recognition," in *Proc. ICASSP*, 2013, pp. 7398-7402.

[6] G. Saon, H. Nahamoo, D. Nahamoo and M. Picheny, "Speaker adaptation of neural network acoustic models using i-vectors," in *Proc. ASRU*, 2013, pp. 55-59.

[7] M. Seltzer and J. Droppo, "Multi-task learning in deep neural networks for improved phoneme recognition," in *Proc. ICASSP*, 2013, pp. 6965–6969.

[8] R. Giri, M. Seltzer, J. Droppo, and D. Yu, "Improving speech recognition in reverberation using a room-aware deep neural network and multi-task learning," in *Proc. ICASSP*, 2015, pp. 5014–5018.

[9] A. Narayanan, and D. Wang, "Improving robustness of deep neural network acoustic models via speech separation and joint adaptive training," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 23, no. 1, pp. 92–101, Jan. 2015.

[10] Z. Wang, and D. Wang, "A joint training framework for robust autiomatic speech recognition," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 24, no. 4, pp. 796–806, Apr. 2016.

[11] M. Delcroix, K. Kinoshita, T. Hori, and T. Nakatami, "Context adaptive deep neural networks for fast acoustic model adaptation," in *Proc. ICASSP.*, 2015, pp. 4535–4539.

[12] M. Delcroix, K. Kinoshita, T. Hori, and T. Nakatami, "Context adaptive deep neural networks for fast acoustic model adaptation in noisy conditions," in *Proc. ICASSP.*, 2016, pp. 5270–5274.

[13] Y. Xu, J. Du, L.-R. Dai, and C.-H. Lee, "Dynamic noise aware training for speech enhancement based on deep neural networks," in *Proc. Interspeech*, 2014, pp. 2670-2674.

[14] R. Caruana, "Multitask learning: A knowledge-based source of inductive bias," *Machine Learning*, vol. 28, pp. 41–75, 1997.

[15] T. Kowaliw, N. Bredeche, and R. Doursat, *Growing adaptive machines: combining development and learning in artificial neural networks*, Springer, 2014.

[16] K. H. Lee, S. J. Kang, W. H. Kang, and N. S. Kim, "Two-stage noise aware training using asymmetric deep denoising autoencoder," in *Proc. ICASSP*, 2016, pp. 5765–5769.

[17] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proc. Int. Conf. Artif. Intell. Statist.,* 2011, pp. 315–323.

[18] X. Feng, Y. Zhang, and J. Glass, "Speech feature denoising and dereverberation via deep autoencoders for noisy reverberant speech recognition," in *Proc. ICASSP*, 2014, pp. 1759-1763.

[19] H. G. Hirsch, AURORA-5 experimental framework for the performance evaluation of speech recognition in case of a hands-free speech input in noisy environments Niederrhein Univ. of Appl. Sci., Nov. 2007.

[20] S. Young *et al*., *The HTK book*. Cambridge, U.K.: Cambridge Univ. Eng. Dept., 2006.

[21] C. W. Han, S. J. Kang, and N. S. Kim, "Reverberation and noise robust feature compensation based on IMM," *IEEE Trans. Audio, Speech, Language Process*., vol. 21, no. 8, pp. 1598-1611, Aug. 2013.

[22] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," *CoRR*, vol. abs/1207.0580, 2012.

[23] 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.

[24] G. Hu. (2004) 100 nonspeech environmental sounds. [Online]. Available: http://web.cse.ohiostate.edu/pnl/corpus/HuNonspeech/HuCorpus.htm