# FULLY DNN-BASED MULTI-LABEL REGRESSION FOR AUDIO TAGGING

*Yong Xu*[*]*, Qiang Huang*[†]*, Wenwu Wang, Philip J. B. Jackson, Mark D. Plumbley*

Centre for Vision, Speech and Signal Processing, University of Surrey, UK
{yx0001, q.huang, w.wang, p.jackson, m.plumbley}@surrey.ac.uk

## ABSTRACT

Acoustic event detection for content analysis in most cases relies on lots of labeled data. However, manually annotating data is a time-consuming task, which thus makes few annotated resources available so far. Unlike audio event detection, automatic audio tagging, a multi-label acoustic event classification task, only relies on weakly labeled data. This is highly desirable to some practical applications using audio analysis. In this paper we propose to use a fully deep neural network (DNN) framework to handle the multi-label classification task in a regression way. Considering that only chunk-level rather than frame-level labels are available, the whole or almost whole frames of the chunk were fed into the DNN to perform a multi-label regression for the expected tags. The fully DNN, which is regarded as an encoding function, can well map the audio features sequence to a multi-tag vector. A deep pyramid structure was also designed to extract more robust high-level features related to the target tags. Further improved methods were adopted, such as the Dropout and background noise aware training, to enhance its generalization capability for new audio recordings in mismatched environments. Compared with the conventional Gaussian Mixture Model (GMM) and support vector machine (SVM) methods, the proposed fully DNN-based method could well utilize the long-term temporal information with the whole chunk as the input. The results show that our approach obtained a 15% relative improvement compared with the official GMM-based method of DCASE 2016 challenge.

***Index Terms***— Audio tagging, deep neural networks, multi-label regression, dropout, DCASE 2016

## 1. INTRODUCTION

Due to the use of smart mobile devices in recent years, huge amounts of multimedia data are generated and uploaded to the internet everyday. These data, such as music, field sounds, broadcast news, and television shows, contain sounds from a wide variety of sources. The need for analyzing these sounds has been now increased as it is useful, e.g., for automatic tagging in audio indexing, automatic sound analysis for audio segmentation or audio context classification. Although supervised approaches have proved to be effective in many applications, their effectiveness relies heavily on the quantity and quality of the training data. Moreover, manually labeling a large amount of data is very time-consuming. To handle this problem, two types of methods have been developed. One is to convert low-level acoustic features into "bag of audio words" using unsupervised learning methods [1, 2, 3, 4, 5]. The second type of methods is based on only weakly labeled data [6], e.g. audio tagging. It is clear that tagging audio chunks needs much less time compared to precisely locating event boundaries within recordings. This will certainly improve tractability of obtaining manual annotations for large databases. In this paper, we will focus on the audio tagging task.

To overcome the lack of annotated training data, Multiple Instance Learning (MIL) is proposed in [7] as a variation of supervised learning for problems with incomplete knowledge about labels of training examples. It aims at classifying bags of instances instead of targeting at classifying single instances. Following this work, Andrew *et al.* [8] proposed a new formulation of MIL as a maximum margin problem, which had led to some further work [9, 10, 11, 12, 13] in audio and video processing using weakly labeled data. Mandel and Ellis in [9] used clip-level tags to derive tags at the track, album, and artist granularities by formulating a number of music information related multiple-instance learning tasks and evaluated two MIL based algorithms on them. In [14], Phan *et al.* used event-driven MIL to learn the key evidences for event detection. Recently, [6] also presented a SVM based MIL system for audio tagging and event detection. GMM, as a common model, was used as the official baseline method in DCASE 2016 for audio tagging. More details can be found in [15].

Although the methods mentioned above have led to some useful results in detection and analysis of audio data, most of them ignored possible relationships of any contextual information and only focused on training the model for each single event class independently. To better use the data with weak labels, our work will utilize the whole or almost whole frames of the observed chunk as the input of a fully deep neural network to make a mapping from an audio feature sequence to a multi-tag vector.

Recently, deep learning technologies have obtained great successes in speech, image and video fields [16, 17, 18, 19] since Hinton and Salakhutdinov showed the insights using a greedy layer-wise unsupervised learning procedure to train a deep model in 2006 [20]. The deep learning methods were also investigated for related tasks, like acoustic scene classification [21] and acoustic event detection [22]. And better performance could be obtained in these tasks. For music tagging task, [23, 24] have also demonstrated the superiority of deep learning methods. However, to the best of our knowledge, the deep learning based methods have not been used for environmental audio tagging which is a newly proposed task in D-CASE 2016 challenge based on the CHiME-home dataset [25]. For the audio tagging task, only the chunk-level instead of frame-level labels were available. Furthermore, multiple instances could happen simultaneously, for example, the *child speech* could exist with *TV sound* for several seconds. Hence, a good way is to feed the DNN with the whole frames of the chunk to predict the multiple tags in the output.

In this paper, we propose a fully DNN-based method, which can

[*]This work was supported by the Engineering and Physical Sciences Research Council (EPSRC) of the UK under the grant EP/N014111/1.

[†]The first author and the second author have equal contribution for this paper.

well utilize the long-term temporary information, to map the whole sequence of audio features into a multi-tag vector. The fully neural network structure was also successfully used in image segmentation [26]. To get a better prediction of the tags, a deep pyramid structure is designed with gradually shrinked size of layers. This deep pyramid structure can reduce the non-correlated interferences in the whole audio features while focusing on extracting the robust high-level features related to the target tags. Dropout [27] and background noise aware training [28] are adopted to further improve the tagging performance in the DNN-based framework.

The rest of the paper is organized as follows. In section 2, we will introduce the related work using GMM and SVM based MIL in detail, and depict our DNN based framework in section 3. The data description and experimental setup will be given in section 4. We will show the related results and discussions in section 5, and finally draw a conclusion in section 6.

## 2. RELATED WORK

Two baseline methods compared in our work are briefly summarized below.

### 2.1. Audio Tagging using Gaussian Mixture Models

Gaussian Mixture Models (GMMs) are a commonly used generative classifier. A GMM is parametrized in $\Theta = \{\omega_m, \mu_m, \Sigma_m\}, m = \{1, \cdots, M\}$, where $M$ is the number of mixtures and $w_m$ is the weight of the $m$-th mixture component.

To implement multi-label classification with simple event tags, a binary classifier is built associating with each audio event class in the training step. For a specific event class, all audio frames in an audio chunk labeled with this event are categorized into a positive class, whereas the remaining features are categorized into a negative class. On the classification stage, given an audio chunk $C_i$, the likelihoods of each audio frame $x_{ij}, (j \in \{1 \cdots L_{C_i}\})$ are calculated for the two class models, respectively. Given audio event class $k$ and chunk $C_i$, the classification score $S_{C_{ik}}$ is obtained as log-likelihood ratio:

$$S_{C_{ik}} = \sum_j log(f(x_{ij}, \Theta_{pos})) - \sum_j log(f(x_{ij}, \Theta_{neg})) \quad (1)$$

### 2.2. Audio Tagging using Multiple Instance SVM

Multiple instance learning is described in terms of bags $\mathbf{B}$. The $j$th instance in the $i$th bag, $B_i$, is defined as $x_{ij}$ where $j \in I = \{1 \cdots l_i\}$, and $l_i$ is the number of instances in $B_i$. $B_i$'s label is $Y_i \in \{-1, 1\}$. If $Y_i = -1$, then $x_{ij} = -1$ for all $j$. If $Y_i = 1$, then at least one instance $x_{ij} \in B_i$ is a positive example of the underlying concept [8].

As MI-SVM is the bag-level MIL support vector machine to maximize the bag margin, we define the functional margin of a bag with respect to a hyper-plane as:

$$\gamma_i = Y_i \max_{j \in I}(\langle \mathbf{w}, \mathbf{x}_{ij} \rangle + b) \quad (2)$$

Using the above notion, MI-SVM can be defined as:

$$\min_{\mathbf{w}, b, \xi} \frac{1}{2}\|\mathbf{w}^2\| + A\sum_i \xi_i \quad (3)$$

$$\text{subject to:} \quad \forall_i : \gamma_i \geq 1 - \xi_i, \quad \xi_i \geq 0$$

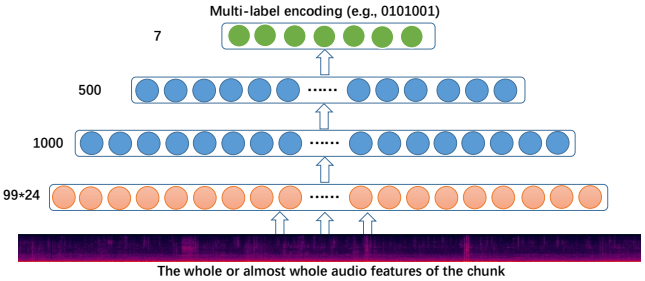


Figure 1: Fully DNN-based audio tagging framework using the deep pyramid structure.

where $\mathbf{w}$ is weight vector, $b$ is bias, $\xi$ is margin violation, and $A$ is a regularization parameter.

Classification with MI-SVM proceeds in two steps. In the first step, $\mathbf{x}_i$ is initialized as the centroid for every positive bag $B_i$ as follows

$$\overline{\mathbf{x}}_i = \sum_{j \in I} \mathbf{x}_{ij} / l_i \quad (4)$$

The second step is an iterative procedure in order to optimize the parameters.

Firstly, $\mathbf{w}$ and $b$ are computed for the data set with positive samples $\{x_I : Y_i = 1\}$.

Secondly, we compute

$$f_{ij} = \langle \mathbf{w}, \mathbf{x}_{ij} \rangle + b, \qquad \mathbf{x}_{ij} \in \mathbf{B}_i$$

Thirdly, we change $\overline{\mathbf{x}}_i$ by

$$\overline{\mathbf{x}}_i = \mathbf{x}_j$$
$$j = \arg\max_{j \in I} f_{ij}, \forall I, Y_I = 1$$

The iteration in this step will stop when there is no change of $\overline{\mathbf{x}}_i$, The optimized parameters will be used for test.

## 3. PROPOSED FULLY DNN-BASED AUDIO TAGGING

DNN is a non-linear multi-layer model for extracting robust features related to a specific classification [18] or regression [17] task. The objective of the audio tagging task is to perform multi-label classification on audio chunks (i.e. assign zero or more labels to each audio chunk of a length e.g. four seconds in our experiments). This chunk only has utterance-level labels without frame-level labels. Multiple events happen at many particular frames. Hence, the common frame-level cross entropy based loss function can not be adopted. We propose a method to encode the whole or almost whole chunk.

### 3.1. Fully DNN-based multi-label regression using sequence to sequence mapping

Fig. 1 shows the proposed fully DNN-based audio tagging framework using the deep pyramid structure. With the proposed framework, the whole or almost whole audio features of the chunk are encoded into a vector with values $\{0, 1\}$ in a regression way. Sigmoid was used as the activation function of the output layer to learn the presence probability of certain events. Minimum mean squared error (MMSE) was adopted as the objective function. A stochastic gradient descent algorithm is performed in mini-batches with mul-

tiple epochs to improve learning convergence as follows,

$$Er = \frac{1}{N}\sum_{n=1}^{N} \|\hat{\mathbf{X}}_n(\mathbf{Y}_{n-\tau}^{n+\tau}, \mathbf{W}, \mathbf{b}) - \mathbf{X}_n\|_2^2 \quad (5)$$

where $Er$ is the mean squared error, $\hat{\mathbf{X}}_n(\mathbf{Y}_{n-\tau}^{n+\tau}, \mathbf{W}, \mathbf{b})$ and $\mathbf{X}_n$ denote the estimated and reference tag vector at sample index $n$, respectively, with $N$ representing the mini-batch size, $\mathbf{Y}_{n-\tau}^{n+\tau}$ being the input audio feature vector where the window size of context is $2*\tau+1$. It should be noted that the input window size should cover the whole or almost whole of the chunk considering that the reference tags are in chunk-level rather than frame-level labels. However, slightly relaxing the window size without covering all of the chunk frames could increase the total training samples for DNN. It can improve the performance in our experiments. $(\mathbf{W}, \mathbf{b})$ denoting the weight and bias parameters to be learned. The updated estimate of $\mathbf{W}^\ell$ and $\mathbf{b}^\ell$ in the $\ell$-th layer, with a learning rate $\lambda$, can be computed iteratively as follows:

$$(\mathbf{W}^\ell, \mathbf{b}^\ell) \leftarrow (\mathbf{W}^\ell, \mathbf{b}^\ell) - \lambda \frac{\partial Er}{\partial(\mathbf{W}^\ell, \mathbf{b}^\ell)},\ 1 \le \ell \le L+1 \quad (6)$$

where $L$ denotes the total number of hidden layers and $L+1$ represents the output layer.

During the learning process where the DNN can be regarded as an encoding function, the audio tags are automatically predicted. Hence the multi-label regression rather than classification can be conducted. Two additional methods are given below to improve the DNN-based audio tagging performance.

### 3.2. Dropout for the over-fitting problem

Deep learning architectures have a natural tendency towards overfitting especially when there is little training data. This audio tagging task only has about four hours training data with imbalanced training data distribution for each type of tag. Dropout is a simple but effective way to alleviate this problem [27]. In each training iteration, the feature value of every input unit and the activation of every hidden unit are randomly removed with a predefined probability (e.g., $\rho$). These random perturbations effectively prevent the DNN from learning spurious dependencies. At the decoding stage, the DNN discounts all of the weights involved in the dropout training by $(1-\rho)$, regarded as a model averaging process [29].

A Mismatch problem may also exist in this task, and testing audio segments could be totally different from existed training audio segments due to the presence of lots of background noise. Thus Dropout should be adopted to improve its robustness to generalize to variation in testing segments.

### 3.3. Background noise aware training

Different types of background noise in different recording environments could lead to the mismatch problem between the testing chunks and the training chunks. To alleviate this, we propose a simple background noise aware training (or background noise adaptation method). To enable this noise awareness, the DNN is fed with the primary audio features augmented with an estimate of the background noise. In this way, the DNN can use additional on-line background noise information to better predict the expected tags. The background noise is estimated as follows:

$$\mathbf{V}_n = [\mathbf{Y}_{n-\tau}, ..., \mathbf{Y}_{n-1}, \mathbf{Y}_n, \mathbf{Y}_{n+1}, ..., \mathbf{Y}_{n+\tau}, \hat{\mathbf{Z}}_n] \quad (7)$$

$$\hat{\mathbf{Z}}_n = \frac{1}{T}\sum_{t=1}^{T} \mathbf{Y}_t \quad (8)$$

where the background noise $\hat{\mathbf{Z}}_n$ is fixed over the utterance and estimated using the first $T$ frames. Although this noise estimator is simple, a similar idea was shown to be effective in DNN-based speech enhancement [17, 28].

## 4. EXPERIMENTAL SETUP AND RESULTS

### 4.1. DCASE2016 data set for audio tagging

The data that we used for evaluation is the dataset of Task4 of DCASE 2016 [15]. The audio recordings are made in a domestic environment. The audio data are provided as 4-second chunks at two sampling rates (48kHz and 16kHz) with the 48kHz data in stereo and with the 16kHz data in mono. The 16kHz recordings were obtained by downsampling the right-hand channel of the 48kHz recordings. Each audio file corresponds to a single chunk [15].

For each chunk, multi-label annotations were first obtained from each of the 3 annotators. The annotations are based on a set of 7 label classes as shown in Table 1. A detailed description of the annotation procedure is provided in [25]. To reduce uncertainty of the test data, the evaluation is based on those chunks where 2 or more annotators agreed about label presence across label classes. Moreover, with the aim of approximating typical recording capabilities of commodity hardware, only the monophonic audio data sampled at 16kHz are used for test.

Table 1: Labels used in annotations.

| Label | Description |
|---|---|
| b | Broadband noise |
| c | Child speech |
| f | Adult female speech |
| m | Adult male speech |
| o | Other identifiable sounds |
| p | Percussive sounds, e.g. crash, bang, knock, footsteps |
| v | Video game/TV |

### 4.2. Experimental Setup

In our experiments, following the original configuration of Task4 of DCASE 2016 [15], we use the same five folds as the evaluation set from the given development dataset, and use the remain of the audio recordings for training.

We pre-process each audio chunk by segmenting them using a (80ms) sliding window with a 40ms hop size, and converting each segment into 24-D MFCCs. For each 4-second chunk, 99 frames of MFCCs are obtained. A 91-frame expansion as the input instead of the total frames were found better because this relaxed input scheme can increase the total training samples. Hence the input size of DNN was 2208 with 91-frame MFCCs and also the appended noise vector. One hidden layer with 1000 units and the second hidden layer with 500 units were used to construct a pyramid structure. Seven sigmoid outputs were adopted to predict the seven tags. The learning rate was 0.005. Momentum was set to be 0.9. The dropout rates for input layer and hidden layer were 0.1 and 0.2, respectively. The mini-batch size was 3. $T$ in Equation 8 was 6. It should be noted that the remaining 2432 chunks without 'strongly agreement' labels

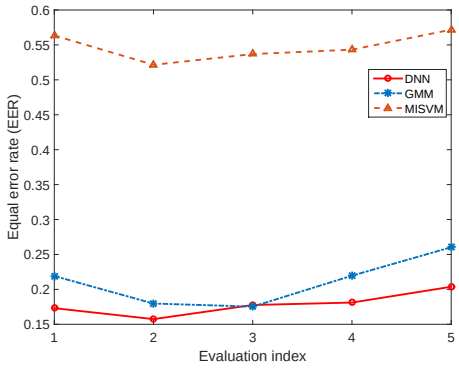


Figure 2: Equal error rates obtained using the proposed fully DNN based approach and the two baselines, namely GMM and MI-SVM across five evaluations folds.

in the development dataset were also added into the DNN training considering that DNN has a better fault-tolerant capability. Meanwhile, these 2432 chunks without 'strongly agreement' labels were also added into the training data for GMM and SVM training.

For a comparison, we also ran two baselines using GMMs and the MI-SVM mentioned in Section 2. For the GMM based method, the number of mixture components is 8. Since the GMM based baseline focuses on computing frame-level likelihoods and MI-SVM prefers to instance-level scores, the sliding window and hop size set for the two baselines are different. The GMM based baseline uses a 20ms sliding window with 10ms hop size, while the sliding window and hop size for MI-SVM are set to be 400ms and 200ms, respectively. To handle audio tagging with MI-SVM, each audio recording will be viewed as a bag and its shorter segments obtained by a sliding window can be treated as an instance. To accelerate computation, we use linear function kernel in our experiments.

To evaluate the effectiveness of our approach, as compared with the two baselines, we use equal error rate (EER) as a metric. EER is defined as the point of the graph of false negative rate (FNR) versus false positive rate (FPR) [30]

$$FNR = \frac{\#false\ negative}{\#positive}$$

$$FPR = \frac{\#false\ positive}{\#negative}$$

EERs are computed individually for each evaluation, and we then average the obtained EERs across the five evaluations to get the final performance.

## 5. RESULTS AND DISCUSSIONS

Figure 2 shows the results obtained using our approach and two baselines. It is clear that the fully DNN-based approach outperforms the two baselines across the five-fold evaluations. This is because of the following two main reasons: First, our proposed approach can well utilize the long-term temporary information instead of treating those information independently. Second, it can map the whole audio features sequence into a multi-tag vector by working as an encoding function. However, GMM and SVM based methods build the models only on single instances. The contextual information and the potential relationship among different tags were not well utilized.

Table 2: Average EER among the proposed fully DNN method, GMM and MI-SVM methods, for each event across five-fold evaluations.

| Various tag | Proposed DNN | GMM | MI-SVM |
|---|---|---|---|
| b | 0.0868 | **0.0755** | 0.1672 |
| c | **0.1686** | 0.2107 | 0.6466 |
| f | **0.2409** | 0.3037 | 0.7626 |
| m | **0.1943** | 0.2847 | 0.7046 |
| o | **0.2867** | 0.2903 | 0.7303 |
| p | **0.2197** | 0.2613 | 0.6724 |
| v | 0.0530 | **0.0484** | 0.1481 |
| Average | **0.1785** | 0.21 | 0.5474 |

The GMM based method yields a close performance to the proposed method only in the third evaluation. We find that two of the audio event classes, namely adult male's speech (label 'm') and other identifiable sounds (label 'o'), are well identified in this fold evaluation. This case is probably because the acoustic characteristics and their variations of the two event classes in the evaluation data can match with the trained models. The use of MI-SVM does not yield competitive performances in comparison with our proposed approach and the GMM-based baseline. This is because MI-SVM, actually working as a discriminative learning, is more sensitive to the quantity and quality of the used training data. Furthermore, MI-SVM does not use the long contextual information.

For a further comparison, Table 2 shows the detailed performances obtained using our approach and the two baselines on each audio tag. We can easily find that the use of the fully-DNN based approach yields great improvements over the two baselines across all of the seven audio tags. Compared with the GMM method, the proposed fully DNN method could get similar performance on tag 'b' and 'v', but it can significantly outperform the competing counterparts on some difficult tags. On average, the proposed DNN method could get a relative 15% improvement by contrasting with the GMM baseline.

## 6. CONCLUSIONS

In this paper we have presented to use a fully-DNN based approach to handle audio tagging with weak labels, in the sense that only the chunk-level instead of the frame-level labels are available. This fully DNN is regarded as an encoding function to map the audio features sequence to a multi-tag vector in a regression way. To extract robust high-level features, a deep pyramid structure was designed to reduce most of the non-correlated interfering features while keeping the highly related features. The dropout and background noise aware training methods were adopted to further improve its generalization capacity for new recordings in unseen environments. We tested our approach on the dataset of the Task4 of the DCASE 2016 challenge, and obtained significant improvements over two baselines, namely GMM and MI-SVM. Compared with the official GMM-based baseline system given in the DCASE 2016 challenge, the proposed DNN system could reduce the EER from 0.21 to 0.1785 on average. For the future work, we will use fully convolutional neural network (CNN) to extract more robust high-level features for the audio tagging task.

## 7. REFERENCES


[1] G. Chen and B. Han, "Improve k-means clustering for audio data by exploring a reasonable sampling rate," in *Seventh International Conference on Fuzzy Systems and Knowledge Discovery*, 2010, pp. 1639–1642.

[2] M. Riley, E. Heinen, and J. Ghosh, "A text retrieval approach to contnent-based audio retrieval," in *Proceedings of International Conference on Music Information Retrieval*, 2008, pp. 1639–1642.

[3] X. Shao, C. Xu, and M. Kankanhalli, "Unsupervised classification of music genre using hidden markov model," in *Proceedings of International Conference on Multimedia and Expo*, 2004, pp. 2023–2026.

[4] R. Cai, L. Lu, and A. Hanjalic, "Unsupervised content discovery in composite audio," in *Proceedings of International Conference on Multimeida*, 2005, pp. 628–637.

[5] T. Sainath, D. Kanevsky, and G. Ivengar, "Unsupervised audio segmentation using extended baum-welch transformations," in *Proceedings of International Conference on Acoustic, Speech and Signal Processing*, 2007, pp. 2009–2012.

[6] A. Kumar and B. Raj, "Audio event detection using weakly labeled data," *CoRR*, vol. abs/1605.02401, 2016. [Online]. Available: http://arxiv.org/abs/1605.02401

[7] T. G. Dietterich, R. H. Lathrop, and T. Lozano-Perez, "Solving the multiple-instance problem with axis-parallel rectangles," *Artificial Intelligence*, vol. 89, pp. 31–71, 1997.

[8] S. Andrew, I. Tsochantaridis, and T. Hofmann, "Support vector machines for multiple-instance learning," in *Proceedings of Advances in Neural Information Processing Systems*, 2003, pp. 557–584.

[9] M. Mandel and D. Ellis, "Multiple-instance learning for music information retrieval," in *The International Society of Music Information Retrieval*, 2008, pp. 577–582.

[10] F. Briggs, B. Lakshminarayanan, L. Neal, X. Z. Fern, and R. Raich, "Acoustic classification of multiple simultaneous bird species: A multi-instance multi-label approach," *Journal of Acoustic Society of America*, vol. 131, pp. 4640–4650, June 2012.

[11] P. Carbonetto, G. Dorko, C. Schmid, H. Kuck, and N. D. Freitas, "Learning to recognize objects with little supervision," *International Journal of Computer Vision*, vol. 77, pp. 219–237, May 2008.

[12] Y. Chen, J. Bi, and J. Z. Wang, "Miles: Multiple-instance learning via embedded instance selection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, pp. 1931–1947, 2006.

[13] A. Ulges, C. Schulze, and T. M. Breuel, "Multiple instance learning from weakly labeled videos," in *Proceedings of the Workshop on Cross-Media Information Analysis, Extraction and Management*, 2008, pp. 17–24.

[14] S. Phan, D. D. Le, and S. Satoh, "Multimedia event detection using event-driven multiple instance learning," in *Proceedings of the 23rd ACM international conference on Multimedia*, 2015, pp. 1255–1258.

[15] http://www.cs.tut.fi/sgn/arg/dcase2016/task-audio-tagging.

[16] 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.

[17] ——, "A regression approach to speech enhancement based on deep neural networks," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 1, pp. 7–19, 2015.

[18] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-R. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath *et al.*, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97, 2012.

[19] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

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

[21] Y. Petetin, C. Laroche, and A. Mayoue, "Deep neural networks for audio scene recognition," in *23rd European Signal Processing Conference (EUSIPCO)*, 2015, pp. 125–129.

[22] E. Cakir, T. Heittola, H. Huttunen, and T. Virtanen, "Polyphonic sound event detection using multi label deep neural networks," in *International Joint Conference on Neural Networks (IJCNN)*, 2015, pp. 1–7.

[23] S. Dieleman and B. Schrauwen, "End-to-end learning for music audio," in *ICASSP*, 2014, pp. 6964–6968.

[24] K. Choi, G. Fazekas, and M. Sandler, "Automatic tagging using deep convolutional neural networks," *arXiv preprint arXiv:1606.00298*, 2016.

[25] P. Foster, S. Sigtia, S. Krstulovic, J. Barker, and M. Plumbley, "CHiME-home: A dataset for sound source recognition in a domestic environment," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, 2015, pp. 1–5.

[26] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3431–3440.

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

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

[29] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," *arXiv preprint arXiv:1207.0580*, 2012.

[30] K. P. Murohy, *Machine Learning: A Probabilistic Perspective*. MIT Press, 2012.