
Proceedings Paper:
Shi, Y., Huang, Q. and Hain, T. orcid.org/0000-0003-0939-3464 (2020) Weakly supervised training of hierarchical attention networks for speaker identification. In: Meng, H., Xu, B. and Zheng, T., (eds.) Proceedings of Interspeech 2020. Interspeech 2020, 25-29 Oct 2020, Shanghai, China. ISCA - International Speech Communication Association , pp. 2992-2996.

https://doi.org/10.21437/interspeech.2020-1774

Weakly Supervised Training of Hierarchical Attention Networks for Speaker Identification

Yanpei Shi, Qiang Huang, Thomas Hain

Speech and Hearing Research Group
Department of Computer Science, University of Sheffield
{YShi30, qiang.huang, t.hain}@sheffield.ac.uk

Abstract

Identifying multiple speakers without knowing where a speaker's voice is in a recording is a challenging task. In this paper, a hierarchical attention network is proposed to solve a weakly labelled speaker identification problem. The use of a hierarchical structure, consisting of a frame-level encoder and a segment-level encoder, aims to learn speaker related information locally and globally. Speech streams are segmented into fragments. The frame-level encoder with attention learns features and highlights the target related frames locally, and output a fragment based embedding. The segment-level encoder works with a second attention layer to emphasize the fragments probably related to target speakers. The global information is finally collected from segment-level module to predict speakers via a classifier. To evaluate the effectiveness of the proposed approach, artificial datasets based on Switchboard Cellular part1 (SWBC) and Voxceleb1 are constructed in two conditions, where speakers' voices are overlapped and not overlapped. Comparing to two baselines the obtained results show that the proposed approach can achieve better performances. Moreover, further experiments are conducted to evaluate the impact of utterance segmentation. The results show that a reasonable segmentation can slightly improve identification performances.

Index Terms: Weakly Supervised Learning, Speaker Identification, Hierarchical Attention, X-vectors, Attention Mechanism

1. Introduction

Speaker identification using deep neural networks becomes an active research area in recent years [1, 2]. In traditional supervised speaker identification training, the data used for training needs hand labelling, where the segments and the corresponding speaker labels are manually annotated [3]. It might be expensive to process a large dataset with a large number of speakers using hand annotation [3, 4].

Instead of hand annotating speaker labels in supervised training, weakly supervised training only relies on the set of speaker labels that occur in the corresponding utterance [5]. This kind of weakly labelled large data collections are available online [3]. Making use of such data collections would be helpful for training with a large amount of data.

Weakly supervised training has been widely used in speech technology. In [3], Karu et.al proposed a DNN based weakly supervised speaker identification training technique. In their work, speaker diarization is firstly applied, and i-vectors are then extracted for each segments. A DNN is trained to predict the set of speaker labels without the true mapping from the i-vectors to the speaker labels. In [6], Xu et,at. proposed a DNN based approach for multi-label audio tagging. In their work, an auto-encoder is trained to predict multiple labels using one input utterance. In [7], Xu et al. proposed to use a gated convolutional neural network for audio classification. In their work, the model is trained to predict one or more classes from an audio without time stamp labels.

Except for speech technology, weakly supervised learning has been widely used in other domains. In [8], Liu et,al. proposed a weakly supervised transfer learning approach to classify multi-temporal remote-sensing images using one labelled image. In [9], Xu et,al. proposed a weakly supervised training approach for image semantic segmentation using image-level labels.

In this work, a hierarchical attention network [10] based weakly supervised speaker identification approach is proposed. In the training and test data, each utterance contains multiple speakers and only the utterance-level labels are available. Different speakers might occur in different part of the input utterance, and some segments might contain multiple overlapped speakers. The model is trained to predict the set of all of the speaker labels from one input utterance [11, 6]. The proposed hierarchical attention network contains a frame-level encoder with attention, and a segment-level encoder with attention, which capture speaker information locally and globally [12]. The frame-level encoder with attention tries to find the important frames within a segment, and the segment-level encoder tries to find the most important parts in the input utterance for speaker identities. Finally, the whole input utterance is compressed into a single vector and input to a DNN classifier. The score vector for each speaker is obtained using a sigmoid function. The proposed hierarchical attention network (HAN) enables the model to highlight and pay attention to the most important parts of input utterance relates the speaker identities.

The rest of the paper is organized as follow: Section 2 presents the architecture of our approach. Section 3 depicts the data and the data construction process, the experimental setup, the baselines to be compared and implementation details. The results are obtained and shown in Section 4, and a conclusion is in Section 5.

2. Proposed Model

Figure 1 shows the architecture of the hierarchical attention network. The network consists of several parts: a frame-level encoder and attention layer, a segment-level encoder and attention layer, and two fully connected layers as a classifier. Given the input acoustic frame vectors, the proposed model applies attention mechanism locally and globally. It predicts multiple speakers in the input utterance. The details of each part will be introduced in the following subsections.

2.1. Frame-Level Encoder and Attention

An utterance is divided into N segments: $\mathbf{S} = \{\mathbf{S}_1, \mathbf{S}_2, \cdots, \mathbf{S}_N\}$ using a sliding window with length M and step H . Each segment $\mathbf{S}_i \in \mathcal{R}^{M \times L} = \{\mathbf{x}_{i1}, \mathbf{x}_{i,2}, \cdots, \mathbf{x}_{i,M}\}$

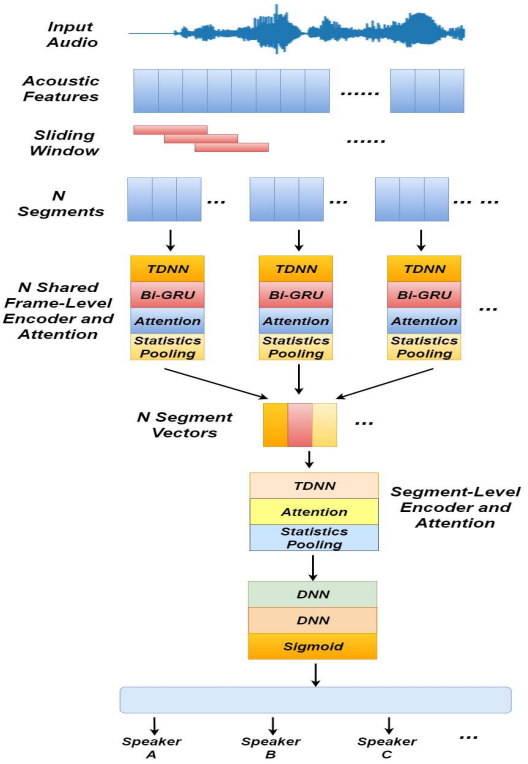

Figure 1: *Architecture of the proposed Hierarchical Attention Network.*

contains M L-dimensional acoustic frame vectors $\boldsymbol{x}_{i,t} \in \mathcal{R}^{1\times L}$, where i denotes the ith segment, t denotes the tth frame, $i \in \{1, \cdots N\}, t \in \{1, \ldots, M\}$.

In the frame-level encoder, a TDNN [13] is used on each segment, and followed by a bidirectional GRU [14] in order to get information from both directions of acoustic frames and contextual information.

The output of a frame-level encoder $\boldsymbol{h}_i = [\overrightarrow{\boldsymbol{h}}_i, \overleftarrow{\boldsymbol{h}}_i] \in \mathcal{R}^{M\times E} = \{\boldsymbol{h}_{i,1}, \boldsymbol{h}_{i,2}, \cdots, \boldsymbol{h}_{i,M}\}$ contains the information of the segment $\boldsymbol{S}_i$.

In the frame-level attention layer, a two-layer MLP is first used to convert $\boldsymbol{h}_i$ into score vector $\boldsymbol{z}_i$, by which a normalized importance weight vector $\boldsymbol{\alpha}_i$ can be computed via a softmax function [10, 15].

$$\alpha_{i,t} = \frac{\exp(z_{i,t})}{\sum_{t=0}^{M} \exp(z_{i,t})} \tag{1}$$

$$z_{i,t} = \text{Relu}(\boldsymbol{h}_{i,t}\boldsymbol{W}_{i,0} + \boldsymbol{b}_{i,0})\boldsymbol{W}_{i,1} \quad , \tag{2}$$

where $z_{i,t}$ and $\alpha_{i,t}$ are a scalar score and normalized score for each time step t respectively. $\boldsymbol{W}_{i,0} \in \mathcal{R}^{E\times E}$, $\boldsymbol{b}_{i,0} \in \mathcal{R}^{1\times E}$ and $\boldsymbol{W}_{i,1} \in \mathcal{R}^{E\times 1}$ are the parameters of a two-layer MLP. These parameters are shared when processing N segments. A weighted output of the frame-level encoder is computed by

$$\boldsymbol{A}_{i,t} = \alpha_{i,t}\boldsymbol{h}_{i,t} \tag{3}$$

Following [16], statistics pooling is applied on $\boldsymbol{A}_i$ to compute its mean vector ($\boldsymbol{\mu}_i$) and std ($\boldsymbol{\sigma}_i$) vector over time. A segment vector $\boldsymbol{V}_{S_i}$ is then obtained by concatenating the two vectors:

$$\boldsymbol{V}_{S_i} = \text{concatenate}(\boldsymbol{\mu}_i, \boldsymbol{\sigma}_i) \tag{4}$$

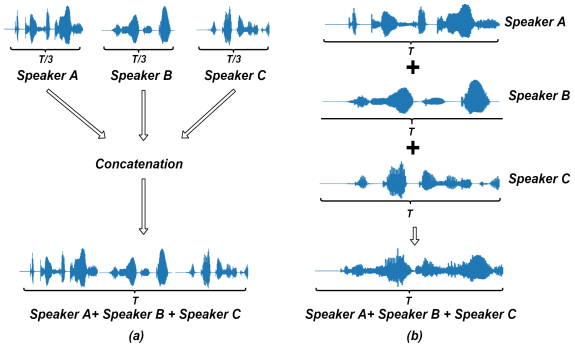

Figure 2: *The illustration of the data construction process. (a): Concat; (b): Overlap.*

2.2. Segment Level Encoder and Attention

For the segment-level encoder and attention, the segment vector sequence is input to a stack of TDNN layers followed by a attention that descript in section 2.1. as the omission of the GRU layer can well accelerate training when processing a large number of samples.

The output of the frame level encoder and attention is $\boldsymbol{V}_S \in \mathcal{R}^{N\times E} = \{\boldsymbol{V}_{S_1}, \boldsymbol{V}_{S_2}, \cdots, \boldsymbol{V}_{S_N}\}$. The weight vector $\boldsymbol{\alpha}^s \in \mathcal{R}^{N\times 1} = \{\alpha_1^s, \alpha_2^s, \cdots, \alpha_N^s\}$ of segment level attention can be computed as follows [17]:

$$\begin{aligned} \alpha_i^s &= \frac{\exp(z_i^s)}{\sum_{i=0}^{N} \exp(z_i^s)} \\ z_i^s &= \text{Relu}(\boldsymbol{V}_{S_i}\boldsymbol{W}_{n,0} + \boldsymbol{b}_{n,0})\boldsymbol{W}_{n,1} \quad , \end{aligned} \tag{5}$$

where z_i^s and α_i^s are a scalar score and normalized score for each segment vector $\boldsymbol{V}_{S_i}$ respectively. $\boldsymbol{W}_{n,0} \in \mathcal{R}^{E\times E}$, $\boldsymbol{b}_{n,0} \in \mathcal{R}^{1\times E}$ and $\boldsymbol{W}_{n,1} \in \mathcal{R}^{E\times 1}$ are the parameters of a two-layer MLP. A vector is generated using a statistics pooling over all weighted segments:

$$\begin{aligned} \boldsymbol{\mu}_U &= mean(\sum_i \alpha_i^s \boldsymbol{S}_i) \\ \boldsymbol{\sigma}_U &= std(\sum_i \alpha_i^s \boldsymbol{S}_i) \\ \boldsymbol{V}_U &= \text{concatenate}(\boldsymbol{\mu}_U, \boldsymbol{\sigma}_U) \end{aligned} \tag{6}$$

The final speaker identity classifier is constructed using a two-layer MLP followed by a sigmoid activation function [18] with $\boldsymbol{V}_U$ being its input. The final speaker identities are the output vector which contains the scores (between 1 and 0) for each speaker. The model is trained using binary cross entropy loss [6]:

$$E_{bce} = -\sum_{n=1}^{N} ||\boldsymbol{Y_n} \log \hat{\boldsymbol{Y}}_{\boldsymbol{n}} + (1 - \boldsymbol{Y_n}) \log(1 - \hat{\boldsymbol{Y}}_{\boldsymbol{n}})|| \tag{7}$$

, where $\hat{\boldsymbol{Y}}_{\boldsymbol{n}}$ denotes the predicted score vector and $\boldsymbol{Y}_n$ denote the reference label vector, N denotes the batch size.

3. Experiments

3.1. Data

In this work, Switchboard Cellular Part 1 (SWBC) [19] and Voxceleb1 [20] dataset are used, as both of them are benchmark datasets and have been widely used in speaker identification. The SWBC dataset contains 130 hours telephone speech with 254 speakers (129 male and 125 female) under various environment conditions. The Voxceleb1 dataset contains 1251 speakers

Name	Original Dataset	Type	#Select Speaker	#Utterance Train	#Utterance Test
SWBC-S	SWBC	Telephone	254	6000	20,000
SWBC-L	SWBC	Telephone	254	100,000	20,000
Vox-S	Voxceleb1	Interview	1000	15000	30,000
Vox-L	Voxceleb1	Interview	1000	150,000	30,000

Table 1: *Details for the construction of the four datasets: SWBC-S, Vox-L, SWBC-S and Vox-L.*

with more than 150,000 utterances collected in the wild. 20-dimensional MFCC [21] is used as the input acoustic features.

3.1.1. Data Construction

As there is no ready-made data for our task, new datasets are conducted manually by using the utterances from the Voxceleb1 and the SWBC dataset. To conduct weakly supervised training, two scenarios are designed: Overlap and Concat. Figure 2 (a) shows an example of the Concat scenario where the three speakers' voices are concatenated without an overlap. Figure 2 (b), shows an example of Overlap scenario where the three speakers' voices are completely overlapped.

Based on the two scenarios above, in order to test the robustness of the proposed approach, for each of the two scenarios, four datasets are generated based on SWBC and Voxceleb1. Table 1 shows the details of the four datasets. For the first dataset (SWBC-S, "S" represents small), SWBC dataset is used and each speaker occurs 30 times in the training set averagely. "SWBC-L" ("L" represents large) contains more training data, each speaker occurs 200 times in the training data averagely, while the amount of the test data keeps the same. The small and large version of the datasets are used to test the robustness of the proposed model in small and large training data. Similar to the configurations in the SWBC based datasets, the datasets that based on Voxceleb1 also have small and large scenarios. In "Vox-S", 1000 speakers are randomly selected from the Voxceleb1 dataset. Each speaker occurs 30 times in the training set. In "Vox-L" dataset, each speaker occurs 300 times in the training set, while the test set is the same as "Vox-S". For each of the eight datasets, the number of speakers in each utterance is randomly chosen from one to three in all of the datasets.

3.2. Experiment Setup

The proposed model is compared with two baselines: X-vectors [16] and Attentive X-vector (Att-Xvector) [22, 23, 2, 24]. X-vectors contains TDNN based frame-level feature extractor, statistics pooling and DNN based segment-level feature extractor. Att-Xvectors uses an global attention mechanism after the TDNN based frame-level feature extractor. The proposed approach is denoted as "H-vector" and it is split into to scenarios: H-vector+sliding window and H-vector+static window. In H-vector+sliding window, the window length M is set to 20 frames, and the step length H is set to 10 frames. In H-vector+static window, the M is set to 20 frames, and the H is set to the same as M, which means there is no overlap for each local segments.

In table 1, each of the four datasets contains two scenarios (Concat and Overlap). In the training process, for all of the eight datasets, the number of speakers in the generated utterances is not fixed, changing from one to three. When the number of speakers is one, the generated utterance is the same as the original utterance. When the number of speakers are two or three, the output utterance contains multiple speakers with or without overlap.

There are no overlaps between the training and test data. The length of all of the generated utterances are fixed at five seconds.

3.3. Evaluation Metric

In this work, equal error rate (EER) [25, 26] is used as the evaluation metric, as it is widely used in multi-label audio tagging [6]. The EER is defined as the point when the false negative (FN) equals to the false positive rate (FP) rate. EER is computed for each individual input and averaged across the whole test set [25].

3.4. Implementation

Level	Model	Input	Output
Frame-Level	TDNN	(M,20)	(M,256)
	Bi-GRU	(M,256)	(M,512)
	Attention	(M,512)	(M,512)
	Statistics Pooling	(M,512)	(1,1024)
Segment-Level	TDNN1	(N,1024)	(N,512)
	TDNN2	(N,512)	(N,512)
	TDNN3	(N,512)	(N,1500)
	Attention	(N,1500)	(N,1500)
	Statistics Pooling	(N,1500)	(1,3000)
Utterance-Level	DNN (512)	(1,3000)	(1,512)
	DNN (K)	(1,512)	(1,K)

Table 2: *Architecture of the proposed hierarchical attention network architecture, where K denotes the total number of speakers.*

Table 2 shows the details of the proposed model architecture. The TDNN in both frame-level and segment-level encoder operates at the current time step. Batch normalizations [27] are added after each layer except for attention layer. Adam optimiser [28] is used for all experiments with $\beta_1 = 0.95$, $\beta_2 = 0.999$, and $\epsilon = 10^{-8}$. The initial learning rate is 10^{-4}.

4. Results

Data Type	Window Size	EER (%)			
		SWBC-S	SWBC-L	Vox-S	Vox-L
Concat	10	12.56	7.15	18.29	13.69
	15	11.87	6.85	18.08	13.34
	20	**11.27**	**6.47**	**17.48**	**13.08**
	25	11.69	6.59	17.81	13.29
	30	12.11	6.92	18.21	13.66
Overlap	10	17.81	15.71	34.37	26.46
	15	16.89	15.05	33.48	25.85
	20	16.24	**14.56**	32.77	**25.39**
	25	**15.99**	15.58	**32.26**	25.94
	30	16.59	16.02	32.86	26.17

Table 3: *The obtained results of the proposed H-vector architecture using different window size M (from 10 to 30 frames), step size H is kept at 10 frames.*

Figure 3 shows the results obtained using the four models (X-vectors, Attentive X-vectors, H-vector with static window and H-vector with sliding window) in different test conditions (1, 2, 3 or multiple speakers) on the eight designed datasets

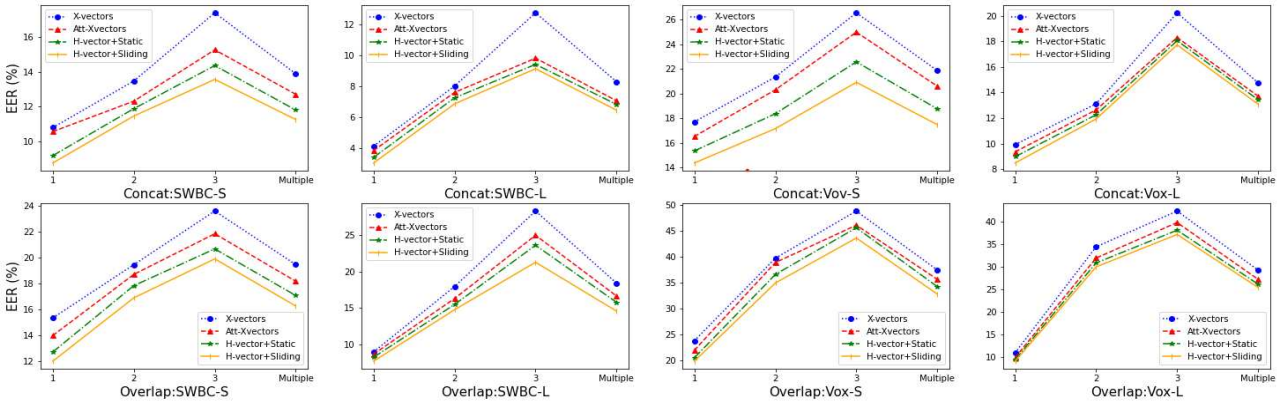

Figure 3: *The results obtained using the four models (X-vectors, Attentive X-vectors, H-vector with static window and H-vector with sliding window) in different test conditions (1, 2, 3 or multiple speakers) on the eight designed datasets (SWBC-S, SWBC-L, Vox-S and Vox-L) and scenarios (Concat and Overlap). For all of the figures, the x-axis represents the number of speakers in test utterance. In H-vector with static window, the window size M is 20 frames. In H-vector with sliding window, the window size M is 20 frames, the step size H is 10 frames.*

Data Type	Step Size	EER (%)			
		SWBC-S	SWBC-L	Vox-S	Vox-L
Concat	5	11.95	6.74	18.01	13.65
	10	**11.27**	6.47	**17.48**	13.08
	15	11.34	**6.29**	17.98	**12.82**
	20	11.45	6.96	18.21	13.15
	25	11.86	6.84	18.56	13.42
Overlap	5	16.49	14.92	33.87	25.51
	10	**16.24**	14.56	**32.77**	25.39
	15	16.88	**14.13**	33.53	**24.86**
	20	17.22	14.82	33.92	25.46
	25	17.78	15.11	34.25	25.81

Table 4: *The obtained results of the proposed H-vector architecture using different step size H (from 5 to 25 frames), window size M is kept at 20 frames.*

(SWBC-S, SWBC-L, Vox-S and Vox-L) and scenarios (Concat and Overlap). In each figure, the X-axis represents the number of speakers in an utterance. “”One, “two”, “three” means the case where an utterance contains only one, two or three speakers, respectively. “Multiple speaker” means the combination of the three cases.

H-vector+Sliding window performs better in almost all of the conditions. The H-vector+static window performs better than the two baselines. These results show that capturing local and global information in weakly supervised speaker identification is helpful. The obtained results by X-vector is worst, this might because it treat each frame has equal importance. Comparing with Att-xvector, one of the reason of the improvement of the proposed H-vector might because of the distributed attention mechanism. Att-Xvector only applied attention mechanism globally.

Among all of the test conditions, the best results are obtained when the number of speakers in each utterance is one, and the worse case is when each utterance contains three speakers. This might due to the difficulty of the test conditions. A similar reason also occurs in the two different data construction scenarios (Concat and Overlap). In these two scenarios, the results obtained on Concat scenario is better than that on Overlap scenario. This might because when the speakers’ voice are overlapped together, it is more difficult to distinguish different speakers. However, the proposed H-vector+sliding window performs better than the baselines in different test conditions and different data construction scenarios.

Moreover, when the training data is small, the proposed H-vector+sliding window still performs better than the baselines and H-vector+static window, reaching 11.5 % and 3.4 % relative improvement than X-vectors and Att-Xvectors in SWBC-S dataset in Concat scenario. It shows robustness of the proposed H-vector+sliding window when there is no enough training data.

In order to test the effectiveness of the window size (M) and step size (H), Table 3 and 4 show the obtained results using the proposed H-vector+sliding window when using different window size and step size. In Overlap scenario, the equal error rate is more sensitive to the change of window size and step size. This might because in Overlap scenario, different speaker signals are overlapped in time domain, some speaker features might influence to each other. Different window size and step size allows the frame-level encoder and attention to capture more local features. Furthermore, in most of the cases, the best results is obtained when the window size is 20 frames, the step size is 10 frames, in which the step size is set to the half-size of the window size.

5. Conclusion and Future Work

In this work, a hierarchical attention network is proposed to solve the weakly labelled speaker identification problem. The input utterance is split into each local segments using a sliding window. Frame-level and segment-level encoder and attention capture speaker information locally and globally. The experiments are done with different test conditions and different amount of training data. The obtained results show that the proposed hierarchical attention network with sliding window performs better than X-vector and Attentive Xvector baselines, as well as hierarchical attention network with static window. In the future work, more complex network architectures and larger dataset such as Voxceleb2 will be investigated.

Acknowledgement

This work was in part supported by Innovate UK Grant number 104264.

6. References

[1] E. Variani, X. Lei, E. McDermott, I. L. Moreno, and J. Gonzalez-Dominguez, "Deep neural networks for small footprint text-dependent speaker verification," in *ICASSP*. IEEE, 2014.

[2] Q. Wang, K. Okabe, K. A. Lee, H. Yamamoto, and T. Koshinaka, "Attention mechanism in speaker recognition: What does it learn in deep speaker embedding?" in *SLT*. IEEE, 2018, pp. 1052–1059.

[3] M. Karu and T. Alumäe, "Weakly supervised training of speaker identification models," in *Proc. Odyssey 2018 The Speaker and Language Recognition Workshop*, 2018, pp. 24–30.

[4] Y. Jia, M. Johnson, W. Macherey, R. J. Weiss, Y. Cao, C.-C. Chiu, N. Ari, S. Laurenzo, and Y. Wu, "Leveraging weakly supervised data to improve end-to-end speech-to-text translation," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 7180–7184.

[5] Z.-H. Zhou, "A brief introduction to weakly supervised learning," *National Science Review*, vol. 5, no. 1, pp. 44–53, 2018.

[6] Y. Xu, Q. Huang, W. Wang, P. Foster, S. Sigtia, P. J. Jackson, and M. D. Plumbley, "Unsupervised feature learning based on deep models for environmental audio tagging," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 6, pp. 1230–1241, 2017.

[7] Y. Xu, Q. Kong, W. Wang, and M. D. Plumbley, "Large-scale weakly supervised audio classification using gated convolutional neural network," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 121–125.

[8] W. Liu, R. Qin, and F. Su, "Weakly supervised classification of time-series of very high resolution remote sensing images by transfer learning," *Remote Sensing Letters*, vol. 10, no. 7, pp. 689–698, 2019.

[9] X. Xu, G. Li, G. Xie, J. Ren, and X. Xie, "Weakly supervised deep semantic segmentation using cnn and elm with semantic candidate regions," *Complexity*, vol. 2019, 2019.

[10] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, and E. Hovy, "Hierarchical attention networks for document classification," in *NAACL*, 2016, pp. 1480–1489.

[11] Z.-L. Zhang and M.-L. Zhang, "Multi-instance multi-label learning with application to scene classification," in *Advances in neural information processing systems*, 2007, pp. 1609–1616.

[12] Y. Shi, Q. Huang, and T. Hain, "H-vectors: Utterance-level speaker embedding using a hierarchical attention model," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 7579–7583.

[13] V. Peddinti, D. Povey, and S. Khudanpur, "A time delay neural network architecture for efficient modeling of long temporal contexts," in *ISCA*, 2015.

[14] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," in *NIPS*, 2014.

[15] M. Rimer and T. Martinez, "Softprop: softmax neural network backpropagation learning," in *2004 IEEE International Joint Conference on Neural Networks (IEEE Cat. No. 04CH37541)*, vol. 2. IEEE, 2004, pp. 979–983.

[16] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust dnn embeddings for speaker recognition," in *ICASSP*. IEEE, 2018, pp. 5329–5333.

[17] Y. Pan, B. Mirheidari, M. Reuber, A. Venneri, D. Blackburn, and H. Christensen, "Automatic hierarchical attention neural network for detecting ad," in *Interspeech*, 2019.

[18] Y. Ito, "Representation of functions by superpositions of a step or sigmoid function and their applications to neural network theory," *Neural Networks*, vol. 4, no. 3, pp. 385–394, 1991.

[19] D. M. David Graff, Kevin Walker, "Switchboard cellular part 1 audio," https://catalog.ldc.upenn.edu/LDC2001S13, 2001.

[20] A. Nagrani, J. S. Chung, and A. Zisserman, "Voxceleb: a large-scale speaker identification dataset," *arXiv preprint arXiv:1706.08612*, 2017.

[21] V. Tiwari, "Mfcc and its applications in speaker recognition," *International journal on emerging technologies*, pp. 19–22, 2010.

[22] Y. Zhu, T. Ko, D. Snyder, B. Mak, and D. Povey, "Self-attentive speaker embeddings for text-independent speaker verification." in *Interspeech*, 2018, pp. 3573–3577.

[23] K. Okabe, T. Koshinaka, and K. Shinoda, "Attentive statistics pooling for deep speaker embedding," *Interspeech*, pp. 2252–2256, 2018.

[24] F. R. rahman Chowdhury, Q. Wang, I. L. Moreno, and L. Wan, "Attention-based models for text-dependent speaker verification," in *ICASSP*. IEEE, 2018, pp. 5359–5363.

[25] J.-M. Cheng and H.-C. Wang, "A method of estimating the equal error rate for automatic speaker verification," in *2004 International Symposium on Chinese Spoken Language Processing*. IEEE, 2004, pp. 285–288.

[26] K. P. Murphy, *Machine learning: a probabilistic perspective*. MIT press, 2012.

[27] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *ICML*, 2015, pp. 448–456.

[28] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2014.