Transfer Learning for Voice Activity Detection: A Denoising Deep Neural Network Perspective

Xiao-Lei Zhang, Ji Wu

Multimedia Signal and Intelligent Information Processing Laboratory,

Tsinghua National Laboratory for Information Science and Technology,

Department of Electronic Engineering, Tsinghua University, Beijing, China.

huoshan6@126.com, wu_ji@tsinghua.edu.cn

Abstract

Mismatching problem between the source and target noisy corpora severely hinder the practical use of the machine-learning-based voice activity detection (VAD). In this paper, we try to address this problem in the transfer learning prospective. Transfer learning tries to find a common learning machine or a common feature subspace that is shared by both the source corpus and the target corpus. The denoising deep neural network is used as the learning machine. Three transfer techniques, which aim to learn common feature representations, are used for analysis. Experimental results demonstrate the effectiveness of the transfer learning schemes on the mismatch problem.

Index Terms: deep learning, domain adaptation, feature learning, transfer learning, voice activity detection.

1. Introduction

Voice activity detectors (VADs) aim to discover speech from its background noises. They are important frontends of modern speech recognition systems [1–3] and speech signal processing systems [4]. Recently, the machine-learning-based VADs [5–9] have received much attention in that they not only can be integrated to the speech recognition systems naturally but also can fuse the advantages of multiple features [10–15] much better than traditional VADs. However, the machine-learning-based VAD is still far from its practical use. One significant problem is that we are not sure whether the VAD model trained in a given source corpus is still powerful in a target corpus which might have a different distribution with the source corpus.

In this paper, we try to deal with the aforementioned problem by a novel learning method – transfer learning. Generally, transfer learning tries to make the model trained with one or multiple source tasks generalizes well on different but related target tasks, so that the performance gap between the source tasks and the target tasks can be lowered. See [16] for an excellent survey on transfer learning. In respect of *different hypothesis* on whether the source data or target data is manually labeled, the transfer learning technologies can be categorized into four groups [16]. This paper focuses on the *domain adaptation* techniques, where the source data is manually labeled and the target data is unlabeled which is a practical scene that the machine-learning-based VAD will meet. In respect of *what*

to transfer, the transfer learning methods can be categorized to three groups – instance transfer, feature transfer, and parameter transfer (i.e. model transfer) [16, 17]. This paper focuses on the feature transfer techniques. Generally, feature transfer tries to learn a low-dimensional feature representation that is shared by both the source data and the target data, so that the classifier trained on the source data with the learned subspace can generalize well on the target data. The main contributions of this paper are summarized as follows:

1. **Towards the mismatching problem of the machine-learning-based VAD.** We have conducted an extensive experiment from the domain adaptation perspective for the mismatching problem. The recently proposed denoising deep neural network (DDNN) [18] is used as the learning machine. Empirical results show that the transfer learning schemes are more powerful than several state-of-the-art VADs when the source and target corpora are relatively similar. The results also demonstrate the promising future of the practical use of the machine-learning-based VADs.

2. **A useful empirical comparison of three feature-based domain adaptation schemes.** We have proposed three domain adaptations for the DDNN-based VAD. Empirical results show that we can pre-train the deep neural networks in an unsupervised manner either with the source data only or with both the source data and the target data together, but the data for all layers' pre-training have to be the same without interference, which manifested the powerfulness of the pre-training scheme proposed by Hinton *et al.* [19].

We have to note that the main purpose of this paper is to discuss the effectiveness of the transfer learning for the mismatching problem between the source data and the target data but not to propose a specific VAD algorithm. To make the machine-learning-based VAD work well in practice, a lot of efforts are still needed. As an example, the DDNN-based VAD needs the clean speech signals of its noisy speech corpus in the unsupervised pre-training stage, which is an ideal situation.

The remainder of the paper is organized as follows. In Section 2, we first review the recently proposed DDNN and then present three feature-based domain adaptation schemes for the DDNN-based VAD. In Section 3, we present the related work. In Section 4, we conduct an extensive experimental comparison. In Section 5, we conclude this paper with some future work.

Scheme 1 A successful scheme.

1: Pre-train all layers of DDNN with only the target corpus $\mathcal{X}^{(t)}$.
2: Fine-tune the pre-trained DDNN with only the labeled source corpus, i.e. $\mathcal{X}^{(s)} \times \mathcal{Y}^{(s)}$.

Scheme 2 A successful scheme.

1: Take the source corpus $\mathcal{X}^{(s)}$ and the target corpus $\mathcal{X}^{(t)}$ together as a large corpus.
2: Pre-train all layers of DDNN with the large corpus.
3: Fine-tune the pre-trained DDNN with only the labeled source corpus, i.e. $\mathcal{X}^{(s)} \times \mathcal{Y}^{(s)}$.
---

2. Domain Adaptation for VAD

In this section, we first review the denoising deep neural network (DDNN), and then propose three feature-based domain adaptation schemes for the DDNN-based VAD.

2.1. Review: Denoising Deep Neural Network for VAD

DDNN [18] is a deep neural network. It was motivated from the stacked denoising autoencoder [20, 21]. Compared to the deep-belief-network-based VAD [22], it has achieved a success on the performance of the deep layers over shallower layers. The key idea of DDNN is to first minimize the *reconstruction cross-entropy loss* between the noisy speech signal and its corresponding clean speech signal in an unsupervised greedy layer-wise pre-training way, and then fine-tune the entire deep neural network by minimizing the cross-entropy loss between the noisy speech signal and its manual labels for the minimum classification error. One special point of DDNN is that, in the pre-training phase, DDNN needs to train an accompanying deep neural network, i.e. a deep network that tries to reconstruct the clean speech signal from the clean speech signal. This is mainly to supply the noisy speech its optimization objective in each layer.

From the aforementioned, we can see that one weakness of DDNN is that the noisy speech signal needs its corresponding clean speech signal in the pre-training phase. Because this paper focuses on the effectiveness of the transfer learning, this weakness does not hinder the main contributions of the paper.

2.2. Preliminary of the Feature-Based Domain Adaptation

Suppose we have a labeled source corpus $\mathcal{X}^{(s)} \times \mathcal{Y}^{(s)}$, and an unlabeled target corpus $\mathcal{X}^{(t)}$, where $\mathcal{X}$ denotes the acoustic feature corpus and $\mathcal{Y}$ denotes the set of the manual labels. The corpora $\mathcal{X}^{(s)}$ and $\mathcal{X}^{(t)}$ might be sampled from different noise scenarios. The feature-based domain adaptation scheme aims to find a mapping function $\phi(\cdot)$ such that the distribution difference between $\phi\left(\mathcal{X}^{(s)}\right)$ and $\phi\left(\mathcal{X}^{(t)}\right)$ is minimized. Therefore, if we minimize the classification error on the source corpus with the new feature representation, i.e. $\phi\left(\mathcal{X}^{(s)}\right)$, we can also expect to minimize the classification error on the target corpus.

2.3. Domain Adaptation Via Deep Feature Extraction

For the DDNN-based VAD, a number of training schemes for $\phi(\cdot)$ can be developed. The core idea of the development is to first pre-train DDNN in different unsupervised ways and fine-tune DDNN with the labeled source corpus, i.e. $\mathcal{X}^{(s)} \times \mathcal{Y}^{(s)}$, for the minimum classification error.

Scheme 3 A failed scheme.

Input: The desired depth of DDNN, denoted as L, (i.e. the hidden-layer number).
1: **Source DDNN pre-training:** Pre-train the source DDNN with a depth of $L-1$ with only $\mathcal{X}^{(s)}$. The pre-trained source DDNN is denoted as $\left\{\mathbf{W}_l^{(s)}\right\}_{l=1}^{L-1}$. /*Note: this model needs to be trained only once, and used repeatedly for different target corpus.*/
2: **Target DDNN pre-training:** Pre-train the source DDNN with a depth of $L-1$ with only $\mathcal{X}^{(t)}$. The pre-trained target DDNN is denoted as $\left\{\mathbf{W}_l^{(t)}\right\}_{l=1}^{L-1}$.
3: **Hybrid pre-training of the top layer**: Group the output features of the source DDNN and target DDNN together to a large set, and pre-train the L-th layer of DDNN with the large set. The pre-trained model is denoted as $\mathbf{W}_L^{(t)}$.
4: **if** Scheme $3^{(t)}$ **then**
5:     Fine-tune the pre-trained $\left\{\left\{\mathbf{W}_l^{(t)}\right\}_{l=1}^{L-1}, \mathbf{W}_L^{(t)}\right\}$ with only the labeled source corpus, i.e. $\mathcal{X}^{(s)} \times \mathcal{Y}^{(s)}$.
6: **else if** Scheme $3^{(s)}$ **then**
7:     Fine-tune the pre-trained $\left\{\left\{\mathbf{W}_l^{(s)}\right\}_{l=1}^{L-1}, \mathbf{W}_L^{(t)}\right\}$ with only the labeled source corpus, i.e. $\mathcal{X}^{(s)} \times \mathcal{Y}^{(s)}$.
8: **end if**
9: Output the fine-tuned network as the learned DDNN.

In this paper, we present three unsupervised pre-training schemes, which are described in Schemes 1, 2, and 3 respectively. The effectiveness and efficiency of the three schemes are analyzed qualitatively as follows:

Scheme 1 only uses the unlabeled target data $\mathcal{X}^{(t)}$ to initialize DDNN. Because it uses only $\mathcal{X}^{(t)}$ for pre-training, it is supposed to be a relatively poor initialization scheme but computationally efficient. Moreover, when $\mathcal{X}^{(t)}$ is small, the initial point of DDNN might be biased and still suffer from overfitting, hence, the network might not be trained well.

Scheme 2 uses both $\mathcal{X}^{(s)}$ and $\mathcal{X}^{(t)}$ for initializing DDNN, which can learn a good feature representation shared by $\mathcal{X}^{(s)}$ and $\mathcal{X}^{(t)}$. Particularly, when $\mathcal{X}^{(t)}$ is rare, $\mathcal{X}^{(s)}$ can play a sufficient supplementary role to $\mathcal{X}^{(t)}$. Hence, the network is desired to perform gently well on the target test data. However, whenever we meet a new target task, we have to conduct a heavy computation load by training $\mathcal{X}^{(s)}$ and $\mathcal{X}^{(t)}$ jointly in the pre-training phase.

Scheme 3 is designed to be a compromise between Scheme 1 and Scheme 2. Specifically, because we take the supplementary effect of $\mathcal{X}^{(s)}$ merely into the highest layer of DDNN which is a layer that directly influences the performance of DDNN, we might not only transfer the source knowledge to the target domain but also can save a lot of training time, since that 1) the most computationally expensive part of DDNN is the pre-training of the source DDNN which can be trained once for all; 2) the top layer of DDNN usually has much less hidden units (i.e. much less training time) than the bottom modules. Scheme 3 contains two sub-schemes, which is denoted as Scheme $3^{(t)}$ and Scheme $3^{(s)}$ respectively.

Before the experimental section, we emphasize that 1) Schemes 1 and 2 are successful ones because the data for all layers' pre-training are the same, 2) Scheme 3 fails in providing a good initial point for DDNN, because the data for pre-training

all layers is not consistent. The main purpose that we want to share the failed scheme is to tell the critical readers that it provides a compromise thinking between the computationally light Schemes 1 and the computationally heavy Scheme 2, and we might find a successful compromise scheme that is both as effective as Scheme 2 and as efficient as Scheme 1 in the future.

Note that we can use multiple source corpora and multiple target corpora together to train the model freely. But in this paper, we only discuss the empirical performance with one source corpus and one target corpus, leaving the multiple source domain adaptation problem to a future discussion.

3. Related Work

In respect of transfer learning and deep learning, there has been some similar work with the proposed schemes. For example, in [23], Glorot *et al.* adopted a domain adaptation scheme that is exactly the same as Scheme 2 of this paper for the sentiment classification problem. In [24], Collobert and Weston proposed a joint training scheme for the multitask learning problem of natural language processing, whose key idea is similar with Scheme 3. The architecture of [24] is also successfully applied to machine translation [25]. However, Scheme 3 is different from [24, 25] in that our Scheme 3 pre-train the top hidden-layer of DDNN with set $\left\{ \mathcal{X}^{(s)}, \mathcal{X}^{(t)} \right\}$, while the architecture of [24, 25] try to learn a subspace of word mapping in the top-hidden layer with a strong constraint that one word in the lookup table of $\mathcal{X}^{(s)}$ should have a matching word in that of $\mathcal{X}^{(t)}$.

In respect of the VAD study, the distribution difference between different noise scenarios has been mentioned in traditional VADs. For example, in [26], Chang *et al.* used different statistical models for modeling the speech and noise distributions in different noise scenarios. Another related topic with domain adaptation is the online learning methods [27], they update the model parameters according to the historical domain information of the speech signals. Traditional statistical-model-based VADs [26] can also be regarded as unsupervised online learning methods. But to our knowledge, how to combine multiple features effectively is still an open problem in the online learning methods. On the other side, although the domain-adaptation-based VAD works in batch mode, it can combine multiple features effectively and yield a high accuracy without a requirement of heavy manual labeling.

4. Experiments

Seven noisy test corpora of AURORA2 [28] is used for performance analysis. The signal-to-noise ratio level of the audio signals is set to 5 dB. Each test corpus of AURORA2 contains 1001 utterances, which are split randomly into three groups for training, developing and test respectively. Each training set and development set consist of 300 utterances respectively. Each test set consists of 401 utterances.

The sampling rate is 8kHz. We set the frame length to 25ms long with a frame-shift of 10ms. We extract 10 acoustic features from each observation. The detailed information of the features are listed in Table 1. All features are normalized into the range of $[0, 1]$ in dimension.

To simulate the real-world domain adaptation task, we take the training sets of the Street and Subway noise scenarios as two source corpora. For each source corpus, we form 6 domain adaptation tasks by randomly extracting a 30-second audio segment from the training set of each noise type of AURORA2 ex-

Table 1: Features and their attributes. The subscript of each feature is the window length of the feature [29].

ID	Feature	Dimension	ID	Feature	Dimension
1	Pitch	1	7	$MFCC_{16}$	20
2	DFT	16	8	LPC	12
3	DFT_8	16	9	RASTA-PLP	17
4	DFT_{16}	16	10	AMS	135
5	MFCC	20		**Total**	273
6	$MFCC_8$	20			

cept that of the source corpus. For each domain adaptation task, the development set of the source corpus is used for model selection. We run each domain adaptation task 5 times and report the average accuracies.

The parameters are set as follows. Up to three hidden layers are adopted with the numbers of the hidden units set to $[54, 7, 7]$ respectively. The learning rate of the unsupervised pre-training is set to 0.004. The maximum epoch of the unsupervised pre-training is set to 200. The learning rate of the supervised fune-tuning is set to 0.005. The maximum epoch of the supervised fune-tuning is set to 130. The batch mode training is adopted. Each batch contains 512 observations. Note that the parameters are selected empirically for a compromise between the training time and the accuracy. The accuracy might be further improved by tuning the parameters.

To evaluate the effectiveness of the proposed domain adaptation schemes, we give the empirical lower bound and upper bound that the schemes might achieve. The lower bound, denoted as "LB", is obtained by training DDNN with only the source corpus and testing it on various target environments. If the performance of the proposed domain adaptation schemes is worse than LB, it means that the schemes fail. The performance upper bound, denoted as "UB", is obtained by training DDNN with the training set of the target corpus and testing it on the test set of the same target environments. If the performance of the proposed domain adaptation schemes is better than the UB, it means that the schemes achieve unbelievably amazing successes. We also compare with the G.729B VAD [30], ETSI advanced frontend via Wiener filter [31], ETSI advanced frontend via frame dropping [31], Sohn VAD [32], Ramirez05 VAD [29], Ramirez07 VAD [33], Yu VAD [5], Shin VAD [34], and Ying VAD [27]. The experimental settings are exactly as [22] did.

4.1. Experimental Results

First, we give the Hinton diagram of the feature distributions in different noise scenarios in Fig. 1. From the figure, we can see that most feature distributions are relatively similar with each other except the subway noise scenario, which means the transfer learning schemes might be useful.

Table 2 lists the transfer accuracies with the street noise as the source corpus. From the figure, we can see that in all layers, Scheme 2 is the most powerful one, followed by Scheme 1. Both Scheme 2 and Scheme 1 achieve higher accuracies than LB, which means the *positive transfer* [16] phenomenon is observed. However, both Scheme $3^{(t)}$ and Scheme $3^{(s)}$ are not only worse than Schemes 1 and 2, but also sometimes slightly worse than LB, which means the *negative transfer* [16] is observed. This phenomenon is rather important. It manifested empirically that using the greedy layer-wise pre-training to initialize the deep network is valuable. If we interrupt the initial point of some layer by noises, or if the data for pre-training are inconsistent in all layers, the performance drops dramatically.

Table 3 lists the transfer accuracies with the subway noise

Table 2: Transfer accuracy comparison (in percentage) with the street noise corpus (identification = 4) as the source data. "LB" is short for the lower bound, "S1" is short for Scheme 1, "S2" is short for Scheme 2, "S3$^{(t)}$" is short for Scheme 3$^{(t)}$, "S3$^{(s)}$" is short for Scheme 3$^{(s)}$, and "UB" is short for upper bound. "# layers" means that the depth of the DDNN is "#". Because the experimental environment settings are exactly as [22] did, we just copy the results of the referenced VADs from [22]. Due to the length limit, we only report the best performance of the referenced VADs and its corresponding VAD algorithm. The referenced methods that are marked with "*" means that they are machine-learning-based VADs that are trained and tested in the matching environments.

ID	Noise Type	Referenced	1 layer				2 layers						3 layers					
			LB	S1	S2	UB	LB	S1	S2	S3$^{(t)}$	S3$^{(s)}$	UB	LB	S1	S2	S3$^{(t)}$	S3$^{(s)}$	UB
1	Babble	75.51 (Ramirez05)	74.95	**77.15**	76.44	*78.61*	74.09	75.67	76.59	75.73	73.17	*78.85*	72.72	75.53	75.92	73.74	72.84	*79.14*
2	Car	79.25 (G.729B)	81.89	82.91	**83.51**	*86.77*	82.05	82.08	83.17	82.19	81.73	*86.96*	81.49	81.81	82.92	81.11	82.20	*87.09*
3	Restaurant	69.59 (Ramirez05)	74.44	75.14	**75.74**	*83.47*	73.84	75.34	75.61	74.53	74.17	*83.90*	73.25	75.59	75.19	**75.76**	73.31	*83.78*
5	Airport	72.45 (Shin)*	77.35	**77.92**	77.56	*82.34*	77.12	77.82	**77.88**	77.51	77.32	*82.45*	76.73	77.34	**77.86**	77.18	76.96	*82.30*
6	Train	75.26 (G.729B)	80.51	81.69	81.22	*83.88*	80.47	80.64	**82.27**	81.42	80.88	*84.21*	79.70	80.76	81.89	80.50	79.90	*84.25*
7	Subway	73.16 (Ramirez05)	68.19	68.19	69.87	*85.84*	68.35	72.69	76.28	68.22	68.17	*85.62*	68.44	74.49	**76.42**	70.70	68.26	*85.73*

Table 3: Transfer accuracy comparison (in percentage) with the subway noise corpus (identification = 7) as the source data.

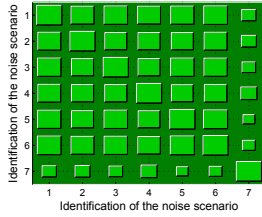| ID	Noise Type	Referenced	1 layer				2 layers						3 layers					
			LB	S1	S2	UB	LB	S1	S2	S3$^{(t)}$	S3$^{(s)}$	UB	LB	S1	S2	S3$^{(t)}$	S3$^{(s)}$	UB
1	Babble	**75.51** (Ramirez05)	54.58	54.60	62.05	*78.61*	54.58	54.59	67.59	54.58	54.58	*78.85*	54.58	54.58	68.11	54.59	54.58	*79.14*
2	Car	**79.25** (G.729B)	55.80	55.80	68.24	*86.77*	59.54	58.05	69.19	63.09	64.11	*86.96*	61.33	57.96	70.05	56.52	58.65	*87.09*

Figure 1: Hinton diagram of the feature distributions in different noise scenarios. Each grid of the Hinton diagram measures the distribution similarity of the features in the relevant two scenarios. The bigger the grid is, the more similar the two distributions are. The similarity is calculated as $\exp\left(-\|\mathbf{c}^{(s)} - \mathbf{c}^{(t)}\|^2/2\right)$ [35] with $\mathbf{c}$ as the feature centroid.

Table 4: Pre-training time (in seconds) comparison.

	1 layer	2 layers		3 layers	
S1	570.48	713.21		774.87	
S2	11200.19	12552.96		12838.95	
S3	–	Source	Hybrid	Source	Hybrid
	–	12055.02	1860.34	12592.76	985.92

as the source corpus. Due to the length limit, we only show the results of two target noise corpora. The experimental phenomena in other noise scenarios are similar with the two. From the table, we can see that due to the significant difference between the subway noise and the target corpus, the accuracies of all schemes drop significantly from UB. However, we can also observe that the accuracies yielded from Scheme 2 are still significantly better than LB and are upgraded layer by layer, which means that the positive transfer is observed too.

As a conclusion, 1) the proposed schemes are effective in dealing with the mismatching problem between the source data and the target data; 2) Schemes 1 and 2 are both effective transfer learning schemes; 3) Initialization via unsupervised greedy layer-wise pre-training is valuable.

Several other interesting phenomena can be observed by comparing Table 2 and Table 3. We can observe that when the feature distributions of the source data and target data are sim-

ilar, the accuracy of the DDNN-based VAD drops slightly with respect to the depth of the network. One possible explanation is that the distributions can be sufficiently covered by the source corpus, so that we can achieve a desired performance with just one hidden layer of DDNN. On the contrary, when the feature distributions are dissimilar, the accuracy increases dramatically with respect to the depth of the network, which demonstrates the power of the transfer learning schemes. However, when compared with the referenced methods, we can observe that when the source and target environments are relatively similar, the DDNN-based VAD outperforms the referenced methods. But when the environments are severely dissimilar, the DDNN-based VAD is weaker than the referenced ones.

Table 4 lists the pre-training time comparison between the schemes. From the table, we can see that Scheme 1 is the most efficient one, and Scheme 3 is slightly slower than Scheme 1.

5. Conclusions

In this paper, we have tried to solve the mismatching problem between the source corpus and the target corpus in the transfer learning perspective, and further tried three DDNN-based domain adaptation schemes for the problem. Experimental results have shown that Schemes 1 and 2 are effective in dealing with the mismatching problem of VAD when compared with the traditional training method, while Scheme 1 is much more efficient than Scheme 2. The results also have shown that the layer-wise pre-training strategy is important for the success of the deep-learning-based transfer learning schemes. Although Scheme 3 is failed, it does provide an attempt on the compromise between the training time and accuracy, and provide a contrary example for showing the effectiveness of the layer-wise pre-training.

Experimental results also have shown that when the source and target corpora are very dissimilar, the performance might be weaker than the referenced methods. For solving this, pre-training with more unlabeled target data, with multiple source domain, and with more hidden layers might be helpful. Moreover, how to make the performance of Scheme 2 more closer to the upper bound, how to accelerate Scheme 2 and meanwhile keep its effectiveness are also what we want to address. We leave these problems as the future work.

6. References

[1] D. Yu and L. Deng, "Deep-structured hidden conditional random fields for phonetic recognition," in *Proc. INTERSPEECH*, 2010, pp. 2986–2989.

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

[3] G. Hinton, L. Deng, D. Yu, G. Dahl, A. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. Sainath *et al.*, "Deep neural networks for acoustic modeling in speech recognition," *IEEE Signal Process. Mag.*, vol. 29, no. 11, pp. 2–17, 2012.

[4] K. Hana and D. L. Wang, "A classification based approach to speech segregation," *The Journal of the Acoustical Society of America*, vol. 99, pp. 1–34, 2012.

[5] T. Yu and J. H. L. Hansen, "Discriminative training for multiple observation likelihood ratio based voice activity detection," *IEEE Signal Process. Lett.*, vol. 17, no. 11, pp. 897–900, 2010.

[6] J. Wu and X. L. Zhang, "Maximum margin clustering based statistical VAD with multiple observation compound feature," *IEEE Signal Process. Lett.*, vol. 18, no. 5, pp. 283–286, 2011.

[7] ——, "Efficient multiple kernel support vector machine based voice activity detection," *IEEE Signal Process. Lett.*, vol. 18, no. 8, pp. 466–499, 2011.

[8] X. L. Zhang and J. Wu, "Linearithmic time sparse and convex maximum margin clustering," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 42, no. 6, pp. 1669–1692, 2012.

[9] Y. Suh and H. Kim, "Multiple acoustic model-based discriminative likelihood ratio weighting for voice activity detection," *IEEE Signal Process. Lett.*, vol. 19, no. 8, pp. 507–510, 2012.

[10] D. L. Wang, "The time dimension for scene analysis," *IEEE Trans. Neural Netw.*, vol. 16, no. 6, pp. 1401–1426, 2005.

[11] D. L. Wang and G. J. Brown, *Computational auditory scene analysis: principles, algorithms and applications*. Wiley-IEEE Press, 2006.

[12] K. Han and D. L. Wang, "Towards generalizing classification based speech separation," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 1, no. 99, pp. 1–27, 2012.

[13] Y. X. Wang, K. Han, and D. L. Wang, "Exploring monaural features for classification-based speech segregation," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 1, no. 99, pp. 1–10, 2012.

[14] K. Hu and D. L. Wang, "An unsupervised approach to cochannel speech separation," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 1, no. 99, pp. 1–10, 2012.

[15] Y. X. Wang and D. L. Wang, "Towards scaling up classification-based speech separation," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. PP, no. 99, pp. 1–23, 2013.

[16] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, 2010.

[17] F. Sha and B. Kingsbury, "Domain adaptation in machine learning and speech processing," in *Tutorial of INTERSPEECH*, 2012, pp. 1–214.

[18] X.-L. Zhang and J. Wu, "Denoising deep neural networks based voice activity detection," in *Proc. 38th IEEE Int. Conf. Acoustic, Speech, Signal Process.*, 2013, pp. 1–5.

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

[20] P. Vincent, H. Larochelle, Y. Bengio, and P. A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proc. 25th Int. Conf. Mach. Learn.*, 2008, pp. 1096–1103.

[21] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P. A. Manzagol, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion," *J. Mach. Learn. Res.*, vol. 11, pp. 3371–3408, 2010.

[22] X. L. Zhang and J. Wu, "Deep belief networks based voice activity detection," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 21, no. 4, pp. 697–710, 2013.

[23] X. Glorot, A. Bordes, and Y. Bengio, "Domain adaptation for large-scale sentiment classification: A deep learning approach," in *Proc. 28th Int. Conf. Mach. Learn.*, 2011, pp. 1–8.

[24] R. Collobert and J. Weston, "A unified architecture for natural language processing: Deep neural networks with multitask learning," in *Proc. 25th Int. Conf. Mach. Learn.* ACM, 2008, pp. 160–167.

[25] T. Deselaers, S. Hasan, O. Bender, and H. Ney, "A deep learning approach to machine transliteration," in *Proc. 4th Workshop Stat. Mach. Translation*, pp. 233–241.

[26] J. H. Chang, N. S. Kim, and S. K. Mitra, "Voice activity detection based on multiple statistical models," *IEEE Trans. Signal Process.*, vol. 54, no. 6, pp. 1965–1976, 2006.

[27] D. Ying, Y. Yan, J. Dang, and F. Soong, "Voice activity detection based on an unsupervised learning framework," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 19, no. 8, pp. 2624–2644, 2011.

[28] D. Pearce, H. Hirsch *et al.*, "The AURORA experimental framework for the performance evaluation of speech recognition systems under noisy conditions," in *Proc. ICSLP'00*, vol. 4, 2000, pp. 29–32.

[29] J. Ramírez, J. C. Segura, C. Benítez, L. García, and A. Rubio, "Statistical voice activity detection using a multiple observation likelihood ratio test," *IEEE Signal Process. Lett.*, vol. 12, no. 10, pp. 689–692, 2005.

[30] A. Benyassine, E. Shlomot, H. Y. Su, D. Massaloux, C. Lamblin, and J. P. Petit, "ITU-T Recommendation G. 729 Annex B: a silence compression scheme for use with G. 729 optimized for V. 70 digital simultaneous voice and data applications," *IEEE Commun. Mag.*, vol. 35, no. 9, pp. 64–73, 1997.

[31] ETSI, "Speech processing, transmission and quality aspects (STQ); distributed speech recognition; advanced front-end feature extraction algorithm; compression algorithms," *ETSI ES*, vol. 202, no. 050.

[32] J. Sohn, N. S. Kim, and W. Sung, "A statistical model-based voice activity detection," *IEEE Signal Process. Lett.*, vol. 6, no. 1, pp. 1–3, 1999.

[33] J. Ramírez, J. Segura, J. Górriz, and L. García, "Improved voice activity detection using contextual multiple hypothesis testing for robust speech recognition," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 15, no. 8, pp. 2177–2189, 2007.

[34] J. W. Shin, J. H. Chang, and N. S. Kim, "Voice activity detection based on statistical models and machine learning approaches," *Computer Speech & Language*, vol. 24, no. 3, pp. 515–530, 2010.

[35] Q. Liu, X. Liao, H. Li, J. R. Stack, and L. Carin, "Semisupervised multitask learning," *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 31, no. 6, pp. 1074–1086, 2009.