INTERSPEECH 2020
October 25–29, 2020, Shanghai, China

Unsupervised Domain Adaptation Under Label Space Mismatch for Speech Classification

Akhil Mathur[1,2], *Nadia Berthouze*[1], *Nicholas D. Lane*[3]
[1]University College London, UK
[2]Nokia Bell Labs, UK
[3]University of Cambridge, UK
akhil.mathur.17@ucl.ac.uk

Abstract

Unsupervised domain adaptation using adversarial learning has shown promise in adapting speech models from a labeled source domain to an unlabeled target domain. However, prior works make a strong assumption that the label spaces of source and target domains are identical, which can be easily violated in real-world conditions. We present AMLS, an end-to-end architecture that performs *Adaptation under Mismatched Label Spaces* using two weighting schemes to separate shared and private classes in each domain. An evaluation on three speech adaptation tasks, namely gender, microphone, and emotion adaptation, shows that AMLS provides significant accuracy gains over baselines used in speech and vision adaptation tasks. Our contribution paves the way for applying UDA to speech models in unconstrained settings with no assumptions on the source and target label spaces.

Index Terms: speech classification, unsupervised domain adaptation, label space mismatch

1. Introduction

Due to the breakthroughs in supervised deep learning, significant progress has been made in speech and audio classification tasks such as spoken keyword detection [1, 2], emotion recognition [3] and ambient sound classification [4]. However, supervised learning models are susceptible to performance degradation if there is a divergence between training and test data distributions, a phenomenon known as *domain shift* [5]. Such domain shifts can be caused by speaker variability (e.g., accents [6, 7]), ambient noise [8, 9, 10], channel distortions [11] or microphone variability [12]. One solution to this problem is to fine-tune the parameters of a pre-trained model using labeled data from the test (or target) domain. In practice, however, *labeled* data is often unavailable or could be expensive to collect in the target domain. These constraints have led to research in *Unsupervised Domain Adaptation (UDA)*, where the goal is to adapt a model from a labeled source domain to an *unlabeled* target domain. For example, [13] and [14] employed adversarial learning to adapt speech emotion recognition models across languages and datasets, and [12] proposed adapting keyword detection models across microphones.

In this paper, we address a fundamental problem in UDA for speech classification tasks, namely *label space mismatch*. Consider a scenario where a model developer has a labeled dataset of spoken keywords (e.g., Siri, Alexa) from US-English accent speakers (source domain), and train a keyword detection classifier with supervised learning. Now, they wish to deploy this classifier for users with French-English accent (target domain) for whom only unlabeled data is available. The accent variability here is an example of *domain shift*, which may cause performance degradation in the target domain. Hence, the developer can employ UDA to adapt the US-accented model to the unlabeled French-accented domain.

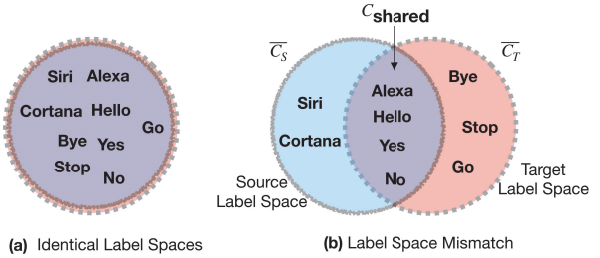

Figure 1: *(a) Identical and (b) Mismatched Label Spaces of Source and Target Domains. The latter is highly likely in real applications and is the focus of our work.*

At their core, most UDA techniques counter domain shift by aligning feature representations of source and target domains. However, they make a strong assumption that the *label spaces of the source and target domains are identical* – in our above example, this would mean that the keyword classes (e.g., Siri, Alexa) must be identical in both source and target datasets (Figure 1(a)). However, as demonstrated in Fig 1(b), this assumption can be easily violated in practice, as the source dataset may contain some keyword classes that are not represented in the target dataset (e.g., Siri, Cortana) and vice versa (e.g., Bye, Stop). We refer to these classes as private classes. This generalized problem setting raises several challenges:

a) **Mitigating negative transfer**. Prior research has shown that the presence of private classes in the dataset can lead to *negative transfer* [15] in adaptation, whereby the adapted classifier performs even worse than a classifier trained solely on the source domain. Note that under a UDA setting, as we have no knowledge of target labels, it is *impossible to know a priori* which classes are private in the target domain or the source domain. Therefore, we need a data-driven solution to identify the shared classes and perform adaptation only between them, while minimizing any negative transfer from private classes.

b) **Labeling private target data as 'unknown'**. After undergoing adaptation, when the classifier is deployed in the target domain, it will encounter target data from shared classes (e.g., Alexa) and private classes (e.g., Stop, Bye). While the classifier can provide labels for the shared class data, it has no knowledge of private class labels (recall that the target domain is unlabeled) – hence, ideally private target data should be classified as 'unknown'. However, prior works show that neural networks tend to output high confidence predictions even for irrelevant or unrecognizable inputs [16, 17, 18]. This, in turn, means that private target instances may get incorrectly labeled as belonging to one of the source classes.

In this work, we first quantify the impact of label space mismatch on speech-based UDA tasks. Then we present our solution, Adaptation under Mismatched Label Spaces (AMLS) wherein our key contribution is in proposing a weighting

http://dx.doi.org/10.21437/Interspeech.2020-1861

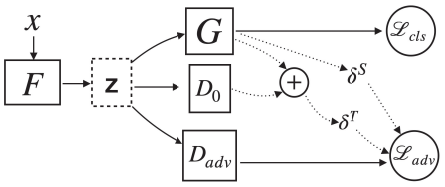

Figure 2: *Architecture of AMLS. Solid boxes represent neural networks and circles denote the losses that are optimized.*

scheme to down-weigh the contribution of private classes and enhance that of shared classes in the adaptation process to counter negative transfer. At the same time, AMLS can accurately identify private target classes and classify them as 'Unknown'. We evaluate AMLS on three tasks: i) gender adaptation in spoken keyword detection, ii) microphone adaptation in spoken keyword detection, and iii) cross-dataset adaptation in speech emotion recognition. Our results show that AMLS outperforms state-of-the-art UDA baselines on all the tasks.

2. Related Work

There is rich literature on training acoustic models robust to real-world variability in speech data. Prior works have studied speaker adaptation using student-teacher learning [19], by doing speaker-adaptive training using i-vectors [20], or by using i-vectors extracted from anchor embeddings [21]. Other works include training noise-robust acoustic models using data augmentation [22] or by layer-wise feature representation matching [23]. Many of these techniques rely on the availability of parallel or labeled data in the target domain.

As acquiring labeled data in a target domain is expensive, there has been an increased focus on using adversarial UDA to adapt acoustic models to target domains, without requiring target labels [24, 25]. Importantly, adversarial UDA techniques also provide a common framework to model different types of adaptation tasks, e.g., cross-lingual emotion adaptation [14], cross-lingual keyword spotting [25], cross-gender ASR [24] and noisy speech adaptation in ASR [26, 27]. We extend these prior works by addressing the critical and previously unaddressed issue of label space mismatch in speech classification tasks.

There have been works in computer vision on label space mismatch, which have addressed the presence of private classes in either source [28] or target domains [29, 30]. Recently, You et al. [31] studied the scenario of label space mismatch for vision tasks. They proposed universal adaptation network (UAN), an architecture that combines prediction entropy and domain similarity into a common metric to separate shared and private classes, and shows promising performance of various vision adaptation tasks. Our work builds on UAN (which we use as a baseline in our experiments) and we show that using our proposed weighting scheme, we can achieve performance improvements over UAN in multiple speech classification tasks.

3. Adaptation under Mismatched Label Spaces (AMLS)

Problem Formulation. We are given a source domain with inputs X_S and labels Y_S sampled from a probability distribution p, and a target domain with inputs X_T sampled from a marginal distribution q. No labels are available from the target domain during training. For consistency with prior work, we denote the label sets of the source and target domains as C_S and C_T respectively. The set of classes shared between source and target domains are denoted by $\mathcal{C}_{\text{shared}} = \mathcal{C}_S \cap \mathcal{C}_T$. Finally, $\overline{C_S}$ = $\mathcal{C}_S \setminus \mathcal{C}_T$ and $\overline{C_T}$ = $\mathcal{C}_T \setminus \mathcal{C}_S$ represent the private label sets of the source and the target domains. Let $p_\cap$ and p_* respectively denote the distribution of source data with label sets $\mathcal{C}_{\text{shared}}$ and $\overline{C_S}$. Let $q_\cap$ and q_* respectively denote the distribution of target data with label sets $\mathcal{C}_{\text{shared}}$ and $\overline{C_T}$. It is worth reiterating that since we have no knowledge of the target labels during training, it is not possible to know C_T, $\mathcal{C}_{\text{shared}}$, $\overline{C_S}$ or $\overline{C_T}$ a priori, as they all depend on the knowledge of target label set.

Our goal is to learn a classifier using domain adaptation which: (i) provides accurate inferences for target data from shared classes $\mathcal{C}_{\text{shared}}$ under the presence of domain shift, (ii) mitigates the negative transfer caused by private classes $\overline{C_S}$ and $\overline{C_T}$ in the adaptation process, and (iii) assigns an 'Unknown' label to data instances from the private target classes $\overline{C_T}$.

Solution. We propose AMLS, an end-to-end architecture for UDA under the presence of label space mismatch. As discussed, the core challenge here comes from the presence of private classes in both domains, which may lead to *negative transfer*. Intuitively, this negative transfer can be mitigated if we can isolate the shared classes $\mathcal{C}_{\text{shared}}$ and only align their feature representations, while ignoring or down-weighing the contribution of the data from private classes during adaptation.

Figure 2 illustrates our proposed architecture for AMLS. Similar to prior works, our solution consists of a feature extractor F, a classifier G and an adversarial discriminator D_{adv}. When an input x is fed to this architecture, a feature representation $z = F(x)$ is obtained. The extracted features are then passed to G to obtain a softmax probability distribution $\hat{y} = G(z)$ over the source labels C_S. The classifier G is trained on source labeled data using a supervised cross-entropy loss as:

$$\mathcal{L}_{\text{cls}} = -\mathbb{E}_{(x_s,y_s)\sim p} \sum_{k=1}^{K} \mathbb{1}_{[k=y_s]} [\log(G(F(x_s))] \quad (1)$$

Next, the task of aligning feature representations of source and target domains is performed by D_{adv} using adversarial learning. In vanilla UDA, D_{adv} would align the representations over the entire label space, however, this can lead to *negative transfer* due to the presence of private classes. One way to mitigate this challenge is to force D_{adv} to give higher importance (or weights) to the shared classes in the feature alignment process as compared to the private classes. Thus, the (weighted) adversarial loss formulation for D_{adv} is:

$$\begin{aligned}\mathcal{L}_{adv} = &-\mathbb{E}_{(x_s,y_s)\sim p}[\delta^{\mathbf{S}}(\mathbf{x_s}) \log(D_{adv}(F(x_s))] \\ &- \mathbb{E}_{x_t\sim q}[\delta^{\mathbf{T}}(\mathbf{x_t}) \log(1 - D_{adv}(F(x_t))] \quad (2)\end{aligned}$$

where δ^S and δ^T are the weights to be assigned to a source and target sample respectively in the adaptation process. We would like to assign higher weights to samples from shared classes and lower weights to private samples. Formally, these criteria are:

$$0 \leq \mathbb{E}_{x_s\sim p_*} \delta^S(x_s) < \mathbb{E}_{x_s\sim p_\cap} \delta^S(x_s) \leq 1 \quad (3)$$

$$0 \leq \mathbb{E}_{x_t\sim q_*} \delta^T(x_t) < \mathbb{E}_{x_t\sim q_\cap} \delta^T(x_t) \leq 1 \quad (4)$$

The key contribution and technical novelty of our work is in proposing a robust technique to estimate δ^S and δ^T.

Estimating Target Weights. When an input x_t from target domain is fed to the classifier G, we get a probability distribution over the source class set C_S in the form of softmax outputs.

$$\hat{y}_t = G(F(x_t))$$

We hypothesize that the classifier G will be more confident in its predictions $\hat{y}_t$ for inputs from the shared classes $\mathcal{C}_{\text{shared}}$ as

compared to those from the private classes $\overline{\mathcal{C}_T}$. This is a reasonable hypothesis because despite the presence of domain shift, classes in $\mathcal{C}_{\text{shared}}$ are likely to be closer to the source domain as compared to the private classes $\overline{\mathcal{C}_T}$. Hence, any measure of prediction confidence that satisfy Equation 4 can be used as a weighting function to separate $\mathcal{C}_{\text{shared}}$ and $\overline{\mathcal{C}_T}$.

We propose to employ Maximum Margin (MM) as a criterion for classifier confidence. Margin Sampling [32] is a well-known technique in active learning to sample data instances for which a classifier is least confident. Formally, Margin M is defined as $M(x_t) = \hat{y}_t^1 - \hat{y}_t^2$, where $\hat{y}_t^1$ and $\hat{y}_t^2$ represent the highest and second-highest softmax outputs in $\hat{y}_t$. When a classifier has high confidence about its top prediction, M will be high. On the contrary, for a data sample for which a classifier is less confident, M, that is the difference between top two softmax outputs will be low. Equation 5 follows from this logic.

$$0 \leq \mathbb{E}_{x_t \sim q_*} M(x_t) < \mathbb{E}_{x_t \sim q_\cap} M(x_t) \leq 1 \tag{5}$$

It is easy to observe that the MM formulation satisfies the target weighting criterion in Eq. 4. However, due to the presence of domain shift, the margins obtained on target data could be noisy and may lead to incorrect weights for target samples. As such, in addition to the margins, we also propose to employ a domain discriminator to separate private and shared target classes. To this end, we train another domain discriminator D_0 to separate samples from source and target domains. More specifically, we assign a label=1 to the source data and label=0 to the target data, and train the discriminator using the Binary Cross-Entropy loss. Once trained, we hypothesize that D_0 would output a higher score to samples from the shared target classes as they have more similarity with the source domain, and hence we can expect the following to hold true when target domain data is fed to D_0.

$$0 \leq \mathbb{E}_{x_t \sim q_*} D_0(x_t) < \mathbb{E}_{x_t \sim q_\cap} D_0(x_t) \leq 1 \tag{6}$$

The use of both margin and discriminator signals to estimate target weights can potentially offset any noise in one of the measurements; as such, we use $\delta^T(x_t) = 0.5 * (M(x_t) + D_0(x_t))$ which indeed satisfies the weighting criterion in Eq. 4.

Estimating Source Weights. To estimate the source weights, we leverage another interesting property of $\hat{y}_t$. Recall that $\hat{y}_t$ is a probability distribution over the source label space C_S, denoting the probability of a *target sample* x_t belonging to different classes in the source set C_S. We hypothesize that source classes $\mathcal{C}_{\text{shared}}$ which are shared with the target domain will be given higher probabilities in $\hat{y}_t$ and the private source classes $\overline{C_S}$ will have lower probabilities. This is reasonable because target data x_t has no overlap with private source classes, hence the classifier should estimate low probabilities for $\overline{C_S}$. Thus, by observing the probability distribution over classes, we can potentially distinguish shared and private source classes and assign appropriate weights to them.

However, due to presence of samples from private target classes, these class probabilities could be noisy. To address this, we use the target weights obtained earlier to calculate instance-weighted class probabilities and average them over an entire batch B of data to obtain the mean class probability vector η.

$$\eta = \frac{1}{|B|} \sum_{i=1}^{|B|} G(F(x_t^i)) * \delta^T(x_t^i) \tag{7}$$

Effectively, η could be interpreted as a $|\mathcal{C}_{\text{shared}}|$-dimensional vector reflecting the relative weight of each source class in a given batch. We set the source weights $\delta^S(x_s) = \eta_{y_s}$ and expect that samples from shared source classes will be assigned higher weights than samples from private source classes. It can be verified that this weighting scheme satisfies the source weighting criterion specified in Eq. 3.

Training Pipeline. Now we are ready to explain the end-to-end training pipeline of AMLS. As with other adversarial training architectures, we jointly optimize the classification loss on the labeled source domain $\mathcal{L}_{\text{cls}}$ along with the weighted adversarial loss $\mathcal{L}_{\text{adv}}$ shown in Equation 2 where the source weights δ^S and target weights δ^T are calculated as described above.

In addition, we also propose to generate pseudo labels [33] for the target data and use supervised learning to further refine the classifier for use in target domain. However, the challenge with using pseudo-labels is that in the presence of label mismatch, it can lead to severe negative transfer if the pseudo-labels are inaccurate. We alleviate this challenge by leveraging the target weights δ^T that were estimated earlier and only perform pseudo-label based supervised training on target samples whose weights are above a certain threshold δ_p. Formally, the pseudo label classification loss can be expressed as follows:

$$\mathcal{L}_{\text{pseudo}} = -\mathbb{E}_{x_t \sim q} \left[\mathbb{1}_{[\delta(x_t) > \delta_p]} \cdot \sum_{k=1}^{K} \mathbb{1}_{[k=\text{argmax}(\hat{y}_t)]} [\log(\hat{y}_t)] \right]$$

Putting them together, the combined optimization objective of AMLS is:

$$\max_{D_{\text{adv}}} \min_{F,G} \mathcal{L}_{\text{cls}} + \lambda \mathcal{L}_{\text{pseudo}} - \mathcal{L}_{\text{adv}}$$

Inference Pipeline. Given a target sample x_t, we first compute its weight $\delta^T(x_t) = 0.5 * (M(x_t) + D_0(x_t))$. If the weight is below a threshold δ_0, it is likely that this sample belongs to a private class and hence we label it as 'Unknown'. Otherwise, we compute $\hat{y}_t = G(F(x_t))$ and output argmax$(\hat{y}_t)$ as its label. Note that λ, δ_0 and δ_p are hyperparameters that are tuned using cross-validation.

4. Evaluation

We now describe our evaluation setup and results.

Tasks and Datasets. AMLS is evaluated on three speech adaptation tasks. (i) *Gender Adaptation*: We study cross-gender adaptation in a Keyword Classification model. For this, we use the Spoken Keywords dataset [34] consisting of >100k speech utterances from 35 keyword classes (e.g., Yes, Right), and partition it based on the speaker's gender (obtained through a crowd-sourced gender-labeling exercise). The partitions, Male (M) and Female(F), represent different domains and we show results for M→F and F→M adaptation. (ii) *Microphone Adaptation*: We use the Mic2Mic dataset [12] which has spoken keyword recordings from 31 classes simultaneously recorded on multiple microphones such as Matrix Voice (M), ReSpeaker (R) and USB (U). Each microphone represents a domain and the task is to adapt a keyword detection model trained on a source microphone to a target microphone. We pick M→R and U→M as the adaptation tasks. (iii) *Dataset Adaptation*: Finally, we evaluate a challenging task of adapting a speech emotion classification model trained on a source dataset (CREMAD [35]) to a target dataset (RAVDESS [36]) collected in a completely different environment. Here, the datasets represent different domains and we evaluate the CREMAD→RAVDESS task.

Model Architectures. In line with prior works [37, 1], we use convolutional neural networks (CNNs) to build the *Keyword Classification* (KC) and *Emotion Classification* (EC) models.

Table 1: *Target domain accuracy averaged over shared (C_{shared}) and private ($\overline{C_T}$) classes. AMLS significantly outperforms ADDA and DANN and also provides gains over state-of-the-art UAN technique.*

	Gender Adaptation		Microphone Adaptation		Emotion
	M→F	**F→M**	**M→R**	**U→M**	**C→R**
Source	41.41	35.04	40.11	42.44	35.2
ADDA	41.54	28.90	39.21	39.29	30.0
DANN	40.13	29.85	40.19	42.33	28.18
UAN	66.13	60.82	66.30	67.20	38.96
AMLS	**73.78**	**64.1**	**69.02**	**68.77**	**41.45**

The inputs to these models are two-dimensional tensors extracted from speech utterances, consisting of time frames on one axis and 40 MFCC features on the other axis. The architectures are as follows: for KC, feature extractor F: [Conv: {64,64,64}], classifier G: [FC: {256, 128}] where Conv represents the number of convolution kernels in each layer and FC denotes the number of units in each hidden layer. For EC, feature extractor F: [Conv: {128, 64,64,64}], classifier G: [FC: {256, 128}]. The architecture for D_{adv} and D_0 is [FC: {128, 128}]. Our system is implemented in TensorFlow 2.0.

Evaluation Protocol. We follow the same evaluation protocol as earlier uDA works [31]: 80% of the unlabeled data from the target domain is used during adaptation, and the adapted model is tested on 20% held-out target test set. We report the average accuracy across all target classes, including the private target classes whose ground truth is set to 'Unknown'.

Baselines. AMLS is compared with four baselines: i) *Source Only* where the target data is tested with the source domain model, without adaptation, ii) *ADDA* [38], iii) *DANN* [39] and iv) UAN [31]. ADDA and DANN are well-known UDA techniques, but do not consider label space mismatch – hence, they serve as representative baselines for existing speech UDA works. UAN is designed for vision tasks to handle label space mismatch and hence is an appropriate state-of-the-art baseline.

Results. For our experiments, we partition the label space into source and target domains in order to simulate label space mismatch. For the Gender Adaptation task on the Spoken Keyword dataset, we assign a class number to each keyword class as per alphabetical order, e.g., the first class in alphabetical order is assigned label='1' and so on. Thereafter, we randomly sample 20 classes from the complete label set without replacement and consider them as source classes C_S. The remaining 15 classes are considered as private target classes $\overline{C_T}$. Next, we randomly sample 10 classes from C_S and consider them as the shared classes C_{shared} between source and target domains. Based on this partitioning, we obtain $\overline{C_S} = \{33, 3, 7, 8, 10, 14, 20, 21, 23, 26\}$, $C_{\text{shared}} = \{4, 5, 9, 11, 12, 15, 17, 24, 27, 28\}$ and remaining classes are in $\overline{C_T}$. In Table 1, we report the target test accuracy for this configuration averaged over shared and private classes. When adapting a keyword model trained on Male speakers to Female speakers (M→F) and vice-versa (F→M), we observe that the accuracy of ADDA and DANN is similar or even lower than the source-only baseline, confirming the occurrence of negative transfer. Both UAN and AMLS significantly boost the target performance, with AMLS providing accuracy gains between 3-7% over UAN. Further, Figure 3 (left) illustrates the relative change in the target domain accuracy of different shared classes after adaptation. We can observe that DANN results in a negative transfer for several classes such as 11 and 12, which is countered by AMLS.

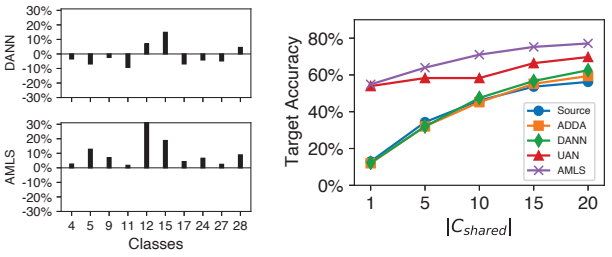

Figure 3: *(left) Relative change in per-class accuracy of shared target classes after adaptation. Severe negative transfer can be observed in DANN. (right) Comparison of different UDA approaches as $|C_{shared}|$ varies. Both experiments are done for the M→F Gender Adaptation task.*

For Microphone Adaptation, we use a similar scheme of partitioning the label space and obtain $\overline{C_S} = \{1, 3, 5, 9, 10, 11, 12, 22, 25, 28\}$, $C_{\text{shared}} = \{6, 8, 14, 16, 21, 23, 24, 27, 29, 30\}$ and remaining classes are in $\overline{C_T}$. From Table 1, we again observe negative transfer during adaptation, as ADDA and DANN perform worse than the source model in some cases. Both UAN and AMLS manage to mitigate negative transfer, with AMLS outperforming UAN by 1.5-3%. Next, for Emotion Adaptation with 7 classes, we use $C_{\text{shared}} = \{\text{Calm, Angry, Fear}\}$, $\overline{C_S} = \{\text{Happy, Sad}\}$ and $\overline{C_T} = \{\text{Disgust, Surprise}\}$. We again observe negative transfer for ADDA and DANN, and AMLS provided 11-13% accuracy gain over them.

Finally, we compare AMLS with the baselines as the size of shared label space increases. We start with the extreme mismatch scenario ($|C_{\text{shared}}| = 1$) and gradually increase the number of shared classes. In Figure 3 (right), we observe that when label space mismatch is high, ADDA and DANN baselines perform poorly due to negative transfer and inability to classify private target classes as 'Unknown'. UAN and AMLS provide significant gains in these scenarios, with AMLS outperforming UAN in all settings. As the label set mismatch reduces, the performances of ADDA and DANN improve significantly, however UAN and AMLS still outperform them. In the other extreme case of no label space mismatch (not shown in the figure), all the adaptation techniques converge to similar accuracies.

5. Conclusion

We presented AMLS, an end-to-end UDA architecture that works under the scenario of label space mismatch, and outperforms existing methods (those used in prior speech works as well in recent computer vision literature) on three adaptation tasks. Our contribution paves the way for speech adaptation algorithms to work in more unconstrained settings by placing no assumption on source and target label spaces.

In future work, we will extend our problem formulation to ASR tasks (e.g., speaker adaptation in ASR) and evaluate the efficacy of AMLS. Future work could also explore how purpose-built features (e.g., speaker i-vectors) can be incorporated to learn robust domain-invariant representations under label space mismatch.

6. References

[1] Y. Zhang, N. Suda, L. Lai, and V. Chandra, "Hello edge: Keyword spotting on microcontrollers," *arXiv preprint arXiv:1711.07128*, 2017.

[2] P. Warden, "Speech commands: A dataset for limited-vocabulary speech recognition," *arXiv preprint arXiv:1804.03209*, 2018.

[3] Y. Xie, R. Liang, Z. Liang, C. Huang, C. Zou, and B. Schuller, "Speech emotion classification using attention-based lstm," *IEEE-ACM TRANSACTIONS ON AUDIO SPEECH AND LANGUAGE PROCESSING*, vol. 27, pp. 1675–1685, 2019. [Online]. Available: http://dx.doi.org/10.1109/TASLP.2019.2925934

[4] D. Stowell, D. Giannoulis, E. Benetos, M. Lagrange, and M. D. Plumbley, "Detection and classification of acoustic scenes and events," *IEEE Transactions on Multimedia*, vol. 17, no. 10, 2015.

[5] J. Quionero-Candela, M. Sugiyama, A. Schwaighofer, and N. D. Lawrence, *Dataset Shift in Machine Learning*. The MIT Press, 2009.

[6] D. Iskra, B. Grosskopf, K. Marasek, H. Heuvel, F. Diehl, and A. Kiessling, "Speecon-speech databases for consumer devices: Database specification and validation," 2002.

[7] T. Viglino, P. Motlicek, and M. Cernak, "End-to-End Accented Speech Recognition," in *Proc. Interspeech*, 2019.

[8] J. Barker, R. Marxer, E. Vincent, and S. Watanabe, "The third 'chime'speech separation and recognition challenge: Dataset, task and baselines," in *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*. IEEE, 2015.

[9] E. Vincent, S. Watanabe, A. A. Nugraha, J. Barker, and R. Marxer, "An analysis of environment, microphone and data simulation mismatches in robust speech recognition," *Computer Speech & Language*, vol. 46, pp. 535–557, 2017.

[10] J. Barker, S. Watanabe, E. Vincent, and J. Trmal, "The fifth'chime'speech separation and recognition challenge: Dataset, task and baselines," in *Interspeech*, 2018.

[11] J. Li, L. Deng, Y. Gong, and R. Haeb-Umbach, "An overview of noise-robust automatic speech recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 4, 2014.

[12] A. Mathur, A. Isopoussu, F. Kawsar, N. Berthouze, and N. D. Lane, "Mic2mic: using cycle-consistent generative adversarial networks to overcome microphone variability in speech systems," in *ACM IPSN 2019*, 2019, pp. 169–180.

[13] M. Abdelwahab and C. Busso, "Domain adversarial for acoustic emotion recognition," *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, vol. 26, no. 12, p. 2423–2435, Dec. 2018.

[14] S. Latif, J. Qadir, and M. Bilal, "Unsupervised adversarial domain adaptation for cross-lingual speech emotion recognition," *CoRR*, vol. abs/1907.06083, 2019.

[15] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on knowledge and data engineering*, vol. 22, no. 10, 2010.

[16] A. Nguyen, J. Yosinski, and J. Clune, "Deep neural networks are easily fooled: High confidence predictions for unrecognizable images," in *CVPR*, 2015, pp. 427–436.

[17] D. Hendrycks and K. Gimpel, "A baseline for detecting misclassified and out-of-distribution examples in neural networks," *arXiv preprint arXiv:1610.02136*, 2016.

[18] S.-M. Moosavi-Dezfooli, A. Fawzi, O. Fawzi, and P. Frossard, "Universal adversarial perturbations," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1765–1773.

[19] J. Li, M. L. Seltzer, X. Wang, R. Zhao, and Y. Gong, "Large-scale domain adaptation via teacher-student learning," *arXiv preprint arXiv:1708.05466*, 2017.

[20] Y. Miao, H. Zhang, and F. Metze, "Speaker adaptive training of deep neural network acoustic models using i-vectors," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 11, pp. 1938–1949, 2015.

[21] Y. Zhao, J. Li, S. Zhang, L. Chen, and Y. Gong, "Domain and speaker adaptation for cortana speech recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5984–5988.

[22] S. Yin, C. Liu, Z. Zhang, Y. Lin, D. Wang, J. Tejedor, T. F. Zheng, and Y. Li, "Noisy training for deep neural networks in speech recognition," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 2015, no. 1, p. 2, 2015.

[23] D. Liang, Z. Huang, and Z. C. Lipton, "Learning noise-invariant representations for robust speech recognition," in *2018 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2018, pp. 56–63.

[24] J.-H. Park, M. Oh, and H.-M. Park, "Unsupervised speech domain adaptation based on disentangled representation learning for robust speech recognition," *arXiv preprint arXiv:1904.06086*, 2019.

[25] J. Hou, P. Guo, S. Sun, F. K. Soong, W. Hu, and L. Xie, "Domain adversarial training for improving keyword spotting performance of esl speech," in *ICASSP*. IEEE, 2019, pp. 8122–8126.

[26] P. Guo, S. Sun, and L. Xie, "Unsupervised adaptation with adversarial dropout regularization for robust speech recognition," 2019.

[27] Q. Wang, W. Rao, S. Sun, L. Xie, E. S. Chng, and H. Li, "Unsupervised domain adaptation via domain adversarial training for speaker recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 4889–4893.

[28] Z. Cao, L. Ma, M. Long, and J. Wang, "Partial adversarial domain adaptation," in *ECCV*. Springer, 2018.

[29] K. Saito, S. Yamamoto, Y. Ushiku, and T. Harada, "Open set domain adaptation by backpropagation," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018.

[30] H. Liu, Z. Cao, M. Long, J. Wang, and Q. Yang, "Separate to adapt: Open set domain adaptation via progressive separation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2927–2936.

[31] K. You, M. Long, Z. Cao, J. Wang, and M. I. Jordan, "Universal domain adaptation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2720–2729.

[32] T. Scheffer, C. Decomain, and S. Wrobel, "Active hidden markov models for information extraction," in *International Symposium on Intelligent Data Analysis*. Springer, 2001, pp. 309–318.

[33] J. Choi, M. Jeong, T. Kim, and C. Kim, "Pseudo-labeling curriculum for unsupervised domain adaptation," *arXiv preprint arXiv:1908.00262*, 2019.

[34] P. Warden, "Speech commands: A dataset for limited-vocabulary speech recognition," *arXiv preprint arXiv:1804.03209*, 2018.

[35] H. Cao, D. G. Cooper, M. K. Keutmann, R. C. Gur, A. Nenkova, and R. Verma, "Crema-d: Crowd-sourced emotional multimodal actors dataset," *IEEE transactions on affective computing*, vol. 5, no. 4, pp. 377–390, 2014.

[36] S. R. Livingstone and F. A. Russo, "The ryerson audio-visual database of emotional speech and song (ravdess): A dynamic, multimodal set of facial and vocal expressions in north american english," *PloS one*, vol. 13, no. 5, 2018.

[37] A. M. Badshah, J. Ahmad, N. Rahim, and S. W. Baik, "Speech emotion recognition from spectrograms with deep convolutional neural network," in *IEEE PlatCon 2017*. IEEE, 2017, pp. 1–5.

[38] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, "Adversarial discriminative domain adaptation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 7167–7176.

[39] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky, "Domain-adversarial training of neural networks," *The Journal of Machine Learning Research*, vol. 17, no. 1, pp. 2096–2030, 2016.