Language/Dialect Recognition Based on Unsupervised Deep Learning

Qian Zhang and John H. L. Hansen

***Abstract*—Over the past decade, bottleneck features within an i-Vector framework have been used for state-of-the-art language/dialect identification (LID/DID). However, traditional bottleneck feature extraction requires additional transcribed speech information. Alternatively, two types of unsupervised deep learning methods are introduced in this study. To address this limitation, an unsupervised bottleneck feature extraction approach is proposed, which is derived from the traditional bottleneck structure but trained with estimated phonetic labels. In addition, based on a generative modeling autoencoder, two types of latent variable learning algorithms are introduced for speech feature processing, which have been previous considered for image processing/reconstruction. Specifically, a variational autoencoder and adversarial autoencoder are utilized on alternative phase of speech processing. To demonstrate the effectiveness of the proposed methods, three corpora are evaluated: 1) a four Chinese dialect dataset, 2) a five Arabic dialect corpus, and 3) multigenre broadcast challenge corpus (MGB-3) for arabic DID. The proposed features are shown to outperform traditional acoustic feature MFCCs consistently across three corpora. Taken collectively, the proposed features achieve up to a relative +58% improvement in C_{avg} for LID/DID without the need of any secondary speech corpora.**

***Index Terms*—Language/Dialect recognition, unsupervised learning, variational autoencoder, adversarial autoencoder, bottleneck feature, phonetic label estimation.**

I. Introduction

In RECENT decades, a number of novel techniques have been proposed to advance language/dialect identification (LID/DID) [1]–[3] or speaker identification (SID) [4]–[6]. Previous studies dedicated on issues related to LID/DID have also been considered, such as code-switching [7], domain mismatch [5], effective out-of-set language rejection [8], [9], etc. Meanwhile, various discriminative modeling techniques have also been investigated, which include both acoustic [10]–[12] and phonetic models[13]. In particular, an i-Vector framework is currently the state-of-the-art discriminative latent feature extraction method which has also been applied on other speech identification tasks [14], [15]. Along with deep neural networks (DNN) introduced into automatic speech recognition, a phonetic-aware DNN has also been proposed for LID [16], [17]. Instead of a Gaussian mixture model (GMM) posterior probability, the output posterior of the DNN senones are used for i-Vector extraction. At the front-end feature level, a variety of robust feature extraction strategies have also been explored [18], addressing different types of channel mismatch or background noise. Subsequently, bottleneck features [19] have been proposed for LID/SID as an alternative feature which contains both acoustic and phonetic information. Stacked bottleneck features have shown dramatic benefits with +45% improvement on the noisy RATS corpus [19]. In addition, [20] shows that bottleneck features trained on the most similar source language perform better than those trained on all available source languages if there are multilingual resources. Recently, Lee *et al.* mentioned in [21] that deep bottleneck features exhibit significant advantages over the shifted delta cepstral (SDC) feature, which has been the predominant option in LRE-11 and its predecessors. Therefore, bottleneck features with an i-Vector framework represent a state-of-the-art strategy for LID.

It is noted, however, an additional transcribed corpus is required for traditional bottleneck feature extraction, since it is based on a well trained DNN based senone recognition system [22] with a bottleneck layer. In addition, usually the transcribed corpus only contains English phonetic information which is less accurate for multiple language forced alignment. There are other potential negative factors between the additional labelled corpus with an original language ID corpus, such as background noise, channel mismatch, speech format mismatch, etc. Actually, either a phonetic-aware DNN based i-Vector or traditional bottleneck feature extraction, which benefits system performance dramatically, utilizes the additional transcribed corpus for phonetic alignment. In other words, the acoustic deep learning strategy are supervised since the senone labels can be learned from manual transcription.

In our study, two strategies are proposed based on an unsupervised deep learning method to boost LID/DID performance. First, an alternative bottleneck feature is proposed based on unsupervised phonetic label estimation. Similar to traditional bottleneck feature, a DNN based phonetic level recognition system is trained with concatenated acoustic input feature. Instead of performing an English senone alignment, a universal phonetic alignment is estimated based on the UBM posterior probability. Therefore, the proposed bottleneck features (i.e., unsupervised

Manuscript received January 15, 2017; revised June 16, 2017 and November 10, 2017; accepted January 3, 2018. Date of publication January 24, 2018; date of current version March 15, 2018. This work was supported by AFRL under contract FA8750-15-1-0205 and in part by the University of Texas at Dallas from the Distinguished University Chair in Telecommunications Engineering held by J. H. L. Hansen. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Najim Dehak. *(Corresponding author: John H. L. Hansen.)*

The authors are with Center for Robust Speech Systems, Erik Jonsson School of Engineering University of Texas at Dallas, Richardson, TX 75080, USA (e-mail: qian.zhang@utdallas.edu; john.hansen@utdallas.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TASLP.2018.2797420

bottleneck features) are extracted without the need of any additional transcribed data. Secondly, two types of autoencoder modeling are introduced for feature extraction, which were previously proposed to extract latent information for image processing. Specifically, variational autoencoder [23] and adversarial autoencoder [24] are considered in our study. Since the autoencoder is used for learning a generative model of data through unsupervised training, it can be utilized at different phases of the system diagram. Meanwhile, the optimal input feature structure for the autoencoder can be found through investigation.

Without loss of generality, there are three corpora used here for algorithm evaluation. First, a relative small dataset with four Chinese dialects is evaluated for all proposed schemes. Since the target Chinese dialect is quite different from each other, it can be considered as a LID task in some sense. Subsequently, a Pan-Arabic corpus which includes 5 nearly spaced dialects, are utilized to evaluate the effectiveness of proposed methods. Both of Chinese and Pan-Arabic corpus have been employed previously for dialect ID system evaluation [25]. In addition, Multi-Genre Broadcast challenge corpus (MGB-3) for Arabic DID is also evaluated in our study.

This paper is organized as follows: Section II describes the proposed bottleneck feature extraction based on unsupervised phonetic label estimation. More theoretical details concerning the two latent variable learning autoencoders (i.e., variational autoencoder and adversarial autoencoder) are presented in Section III. Section 4 illustrates the proposed system framework with alternative autoencoders utilized in different phases. A brief description of the evaluation corpus is included in Section V. The effectiveness of the proposed methods are demonstrated in Section VI through a performance comparison across the three corpora. Finally, conclusions are summarized in Section VII.

II. Bottleneck Feature

A. Traditional Bottleneck Feature Based ASR

Traditional bottleneck features are generally derived from a DNN based ASR acoustic modeling set up with a bottleneck layer. In state-of-the-art ASR systems, each utterance is represented by a sequence of senones (i.e., tied-triphone states) which are introduced for acoustic modeling. Since only word level transcription is usually provided, obtaining senone alignment is the first and fundamental step. Specifically, a hidden Markov model (HMM)/GMM ASR model are employed for forced alignment before subsequent DNN training. A monophone model is trained for acoustic modeling that does not include any contextual information about the preceding or subsequent phone. Based on that, triphone models are created that represent a phoneme variant in the context of two other (left and right) phonemes. Since not all triphones occur in the training data with sufficient statistics, a phonetic decision tree with a maximum likelihood algorithm is used for generating a feasible senone set [26]. In addition, a decision tree is also produced for construction of systems which have unseen triphones. Therefore, senone model training includes additional arguments for the number of HMM states in the decision tree and the number of Gaussians based on heuristics. Once the senone set is defined, Viterbi decoding is employed to align the audio data to the corresponding senones. Usually, the parameters of the acoustic model are estimated in the acoustic training steps which can be better optimized by cycling through training and alignment phases.

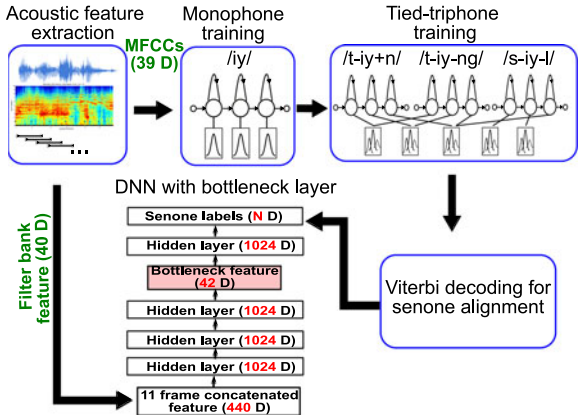

Fig. 1. Traditional bottleneck extraction diagram. ($N = 8973$ for switchboard corpus).

In recent systems, a DNN is used to estimate the senone posteriors based on acoustic features. Instead of MFCCs, the input acoustic feature has been a 40 dimensional filter bank feature. The performance of acoustic modeling is still comparable if we replace one of the layers with a bottleneck layer. Instead of the output layer, traditional bottleneck features are extracted through the middle bottleneck layer. Fig. 1 represents a flow diagram for training a DNN to extract bottleneck features. There are 5 hidden layers (1024-1024-1024-42-1024) between the input layer (e.g., 11 frame concatenated acoustic feature) and output layer (e.g., force aligned senones label). The bottleneck layer is set to be the second to the last layer according to previous work [27], which shows that it contains more discriminative language information compared with other layers.

B. Bottleneck Feature Based on Unsupervised Phonetic Labeling

Traditional bottleneck features have become popular as an alternative to MFCCs for speech tasks such as ASR, SID and LID, since they contain both acoustic and phonetic information. However, there are two potential negative impacts. First, only English phonetic information is considered for forced alignment during the training phase. Since there are more than one language in a regular LID task, unitary phonetic alignment is less accurate. Secondly, the additional corpus used for DNN training may introduce additional challenges, due to mismatch in acoustics characteristics (i.e., channel information, background noise, speech style format (read/spontaneous), etc.). Our previous study shows that for a LID task on a large-scale challenge corpus language recognition evaluation 2015 (LRE15), bottleneck feature extraction based on a Switchboard corpus trained alignment system improved overall system performance by a relative 10%. However, if the AMI meeting corpus is used for acoustic model training and bottleneck extraction, the performance is even worse than using classic MFCCs, confirming the sensitivity of this solution to acoustic mismatch for the DNN alignment phase.

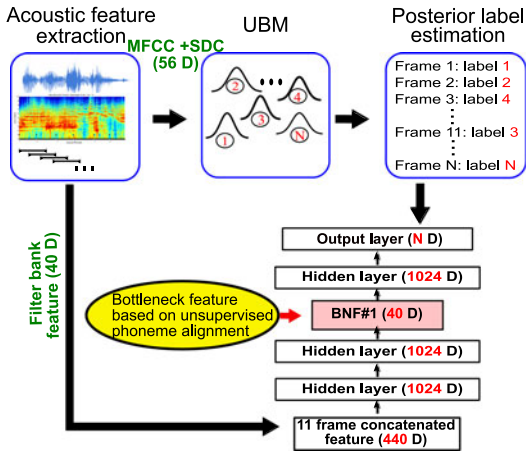

Fig. 2. Unsupervised phonetic label based bottleneck diagram. (N=1024 in our study; explanations for the abbreviations can be found in Section IV).

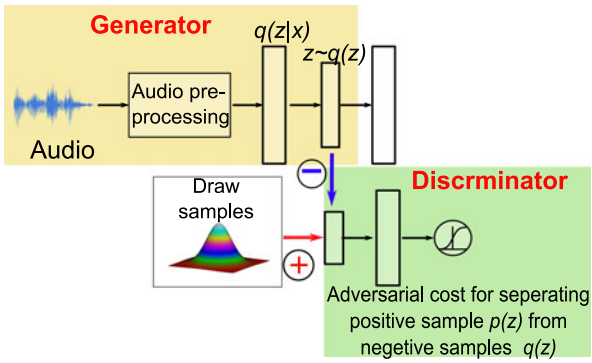

Fig. 3. Adversarial autoencoder.

Therefore, an unsupervised phonetic label based bottleneck feature extraction solution is proposed in this study. The basic concept is similar to traditional bottleneck feature extraction, but without the requirement of an extra transcribed English corpus. Specifically, individual GMM component is adopted for phonetic estimation, because the clusters in acoustic space (e.g., GMM components) mostly reflect phonetics variance. Unlike forced aligned senone labels, the GMM component based phonetic label can be obtained with unsupervised training. The proposed algorithm diagram is shown in Fig. 2. First, a universal background model (UBM) is trained with all enrollment data based on MFCCs with SDC features. Specifically, the universal phonetic space is modelled with N Gaussian mixtures (i.e., N=1024 in our study). Subsequently, frame level phonetic labels are estimated according to posterior probabilities. There are only 4 hidden layers (1024-1024-40-1024) between the input and output layers, because the size of the LID corpora in our study is around 30–50 h, which is smaller than an English ASR corpus (e.g., Switchboard or AMI).

III. Autoencoder

This section introduces the theoretical foundation of autoencoder based latent variable learning models, which includes the variational autoencoder and adversarial autoencoder. They have been popular for unsupervised learning of complicated distributions. In our study, it is the first attempt for speech signal processing, particularly for a LID/DID task.

A. Variational Autoencoder

The observation data set $\boldsymbol{X} = \{\boldsymbol{x}^{(i)}\}_{i=1}^{N}$ consists of N i.i.d samples which are related to an unobserved continuous random variables $\boldsymbol{z}$. Assume there are generative model parameters $\boldsymbol{\theta}$ which determine the distribution of the latent variable $p_{\boldsymbol{\theta}}(\boldsymbol{z})$. In addition, the value $\boldsymbol{x}$ is assumed to be generated according to some conditional probability $p_{\boldsymbol{\theta}}(\boldsymbol{x} \mid \boldsymbol{z})$; with this, the marginal likelihood is $p_{\boldsymbol{\theta}}(\boldsymbol{x}) = \int p_{\boldsymbol{\theta}}(\boldsymbol{z}) p_{\boldsymbol{\theta}}(\boldsymbol{x} \mid \boldsymbol{z})$.

However, the latent value $\boldsymbol{z}$ and true parameter $\boldsymbol{\theta}$ are unknown. In a real scenario, the latent value $\boldsymbol{z}$ is exactly the information which is needed for language recognition given the speech samples $\boldsymbol{x}$. In order to estimate the parameters, the recognition model $q_{\boldsymbol{\phi}}(\boldsymbol{z} \mid \boldsymbol{x})$ (i.e., an approximate of intractable true posterior $p_{\boldsymbol{\theta}}(\boldsymbol{z} \mid \boldsymbol{x})$) is introduced as the probabilistic encoder which generates the distribution of the latent code $\boldsymbol{z}$ given $\boldsymbol{x}$. Similarly, $p_{\boldsymbol{\theta}}(\boldsymbol{x} \mid \boldsymbol{z})$ is referred to as the probabilistic decoder which produces a distribution over the possible corresponding data samples $\boldsymbol{x}$ given the latent variable $\boldsymbol{z}$. Therefore, the marginal likelihood of x, which is composed of a sum over the marginal likelihood of the individual datapoints, is defined as:

$$\begin{aligned}\log p_{\boldsymbol{\theta}}\left(\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(N)}\right) &= \sum_{i=0}^{N} \log p_{\boldsymbol{\theta}}\left(\boldsymbol{x}^{(i)}\right) \\ &= D_{KL}\left(q_{\boldsymbol{\phi}}\left(\boldsymbol{z} \mid \boldsymbol{x}^{(i)}\right) \| p_{\boldsymbol{\theta}}\left(\boldsymbol{z} \mid \boldsymbol{x}^{(i)}\right)\right) \\ &\quad + \mathcal{L}\left(\boldsymbol{\theta}, \boldsymbol{\phi}; \boldsymbol{x}^{(i)}\right). \end{aligned} \tag{1}$$

The first right hand side (RHS) term is the KL divergence of the approximation from the true posterior. Since KL divergence is non-negative, the second RHS term is called the variational lower bound on the marginal likelihood of datapoint i. This term can also be written as,

$$\begin{aligned}\log p_{\boldsymbol{\theta}}\left(\boldsymbol{x}^{(i)}\right) &\geq \mathcal{L}\left(\boldsymbol{\theta}, \boldsymbol{\phi}; \boldsymbol{x}^{(i)}\right) \\ &= \mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})}[-\log q_{\boldsymbol{\phi}}(\boldsymbol{z} \mid \boldsymbol{x}) + \log p_{\boldsymbol{\theta}}(\boldsymbol{x}, \boldsymbol{z})], \end{aligned} \tag{2}$$

which is written as,

$$\begin{aligned}\mathcal{L}\left(\boldsymbol{\theta}, \boldsymbol{\phi}; \boldsymbol{x}^{(i)}\right) &= -D_{KL}\left(q_{\boldsymbol{\phi}}\left(\boldsymbol{z} \mid \boldsymbol{x}^{(i)}\right) \| p_{\boldsymbol{\theta}}(\boldsymbol{z})\right) \\ &\quad + \mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})}[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)} \mid \boldsymbol{z})]. \end{aligned} \tag{3}$$

To minimize the distance between the estimated and true parameters, a lower bound $\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}; \boldsymbol{x}^{(i)})$ needs to be optimized with respect to both the variational parameters $\boldsymbol{\phi}$ and generative parameters $\boldsymbol{\theta}$. Therefore, a stochastic gradient variational Bayes (SGVB) estimator $\widetilde{\mathcal{L}}(\boldsymbol{\theta}, \boldsymbol{\phi}; \boldsymbol{x}^{(i)}) \simeq \mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}; \boldsymbol{x}^{(i)})$ which was previously proposed in [23], is considered since it is a practical estimator with less variance;

$$\begin{aligned}\widetilde{\mathcal{L}}\left(\boldsymbol{\theta}, \boldsymbol{\phi}; \boldsymbol{x}^{(i)}\right) &= -D_{KL}\left(q_{\boldsymbol{\phi}}\left(\boldsymbol{z} \mid \boldsymbol{x}^{(i)}\right) \| p_{\boldsymbol{\theta}}(\boldsymbol{z})\right) \\ &\quad + \frac{1}{L}\sum_{l=1}^{N}\left(\log p_{\boldsymbol{\theta}}\left(\boldsymbol{x}^{(i)} \mid \boldsymbol{z}^{(i,l)}\right)\right), \end{aligned} \tag{4}$$

where $\boldsymbol{z}^{(i,l)} = g_{\boldsymbol{\phi}}(\boldsymbol{\epsilon}^{(l)}, \boldsymbol{x}^{(i)}) \sim q_{\boldsymbol{\phi}}(\boldsymbol{z} \mid \boldsymbol{x}^{(i)})$ and $\boldsymbol{\epsilon}^{(l)} \sim p(\boldsymbol{\epsilon})$. $\boldsymbol{\epsilon}^{(l)}$ is a random noise vector and the differentiable transformation $g_{\boldsymbol{\phi}}(\cdot)$ maps the sample data x with an auxiliary noise vector to a new sample which follows the approximate posterior $q_{\boldsymbol{\phi}}(\boldsymbol{z} \mid \boldsymbol{x}^{(i)})$. With this generative model, the probability of data point x given sample $\boldsymbol{z}^{(i,l)}$ is counted as a negative reconstruction error.

In our study, assuming $p_{\boldsymbol{\theta}}(\boldsymbol{z})$ follows a multivariate Gaussian distribution $\mathcal{N}(\boldsymbol{z}; \boldsymbol{0}, \boldsymbol{I})$, the encoder is a multilayer perception (MLP) with input $\boldsymbol{z}$. Therefore, the true posterior $p_{\boldsymbol{\theta}}(\boldsymbol{x} \mid \boldsymbol{z})$ is also a multivariate Gaussian which is intractable. In this case, the variational approximate posterior, which is assumed to follow a multivariate Gaussian with diagonal covariance, is defined as,

$$\log q_{\boldsymbol{\theta}}\left(\boldsymbol{z} \mid \boldsymbol{x}^{(i)}\right) = \log \mathcal{N}\left(\boldsymbol{z}; \boldsymbol{\mu}^{(i)}, \boldsymbol{\sigma}^{2(i)} I\right), \tag{5}$$

where the mean and standard deviation $\boldsymbol{\mu}^{(i)}$, $\boldsymbol{\sigma}^{(i)}$ can be computed through the weights and bias of the MLP. The resulting estimator for this model is therefore written as,

$$\mathcal{L}\left(\boldsymbol{\theta}, \boldsymbol{\phi}; \boldsymbol{x}^{(i)}\right) \simeq \frac{1}{2}\sum_{j=1}^{J}\left(1 + \log\left(\left(\boldsymbol{\sigma}_j^{(i)}\right)^2\right) - (\boldsymbol{\mu}_j^{(i)})^2 - \left(\boldsymbol{\sigma}_j^{(i)}\right)^2\right) + \frac{1}{L}\sum_{l=1}^{N}\left(\log p_{\boldsymbol{\theta}}\left(\boldsymbol{x}^{(i)} \mid \boldsymbol{z}^{i,l}\right)\right) \tag{6}$$

where $\boldsymbol{z}^{(i,l)} = \boldsymbol{\mu}^{(i)} + \boldsymbol{\sigma}^{(i)} \odot \boldsymbol{\epsilon}^{(l)}$ and $\boldsymbol{\epsilon}^{(l)} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$.

B. Adversarial Autoencoder

The Adversarial autoencoder is a generative model which is trained with dual objectives (i.e., a traditional reconstruction error criterion and an adversarial training criterion [24]). The encoder learns to convert the data distribution to a latent representation with an arbitrary prior distribution. Therefore, the latent information (e.g., phonetic information or the language label) can be extracted from the observation data (e.g., the actual speech) and it will in general follow the desired distribution. The adversarial autoencoder diagram is shown in Fig. 3.

To understand the principle, the generative adversarial networks (GAN) was introduced which establishes a min-max adversarial game between the two neural networks (i.e.,a generative model G and a discriminative model D). The Generator model $G(z)$ maps the samples z from a prior $p(z)$ to the data space. Meanwhile, the discriminator model D computes the posterior probability to distinguish the true samples from fake samples generated from $G(z)$. In addition, $G(z)$ is trained to confuse the discriminator into believing that the generated samples actually follow the data distribution. The solution to this game scenario can be expressed as follows:

$$\min_G \max_D E_{x\sim p_{\mathrm{data}}}[\log D(x)] + E_{z\sim p(z)}[\log(1 - D(G(z)))]. \tag{7}$$

In reality, the purpose of the GAN is to obtain a perfect generator G, whose input is an arbitrary prior and the output follows the same distribution as the true data samples. Based on this concept, an adversarial autoencoder is proposed to force the latent code to follow the desired distribution through the GAN, meanwhile, it attempts to minimize the reconstruction error. In other words, GAN is attached to the latent layer. Here, the negative input of the discriminator D are latent code vectors, and the positive ones are the samples generated according to the prior distribution. Meanwhile, the generator G learns to produce a confusing latent code vector.

Therefore, there are two phases (i.e., reconstruction phase and regularization phase) for adversarial autoencoder training. In the reconstruction phase, the autoencoder updates the encoder's and decoder's parameters in order to obtain a minimum reconstruction error. In the regularization phase, the discriminator's parameters are updated so as to separate the latent code vector (generated from autoencoder) from the desired one (generated from prior distribution). Secondly, the autoencoder updates the encoder parameters to generate potential confusing latent code vectors.

IV. System Diagram

This section focuses on details concerning the baseline and proposed system diagrams based on algorithm and feature discussion in Sections II and III.

Baseline System: In the baseline system, 13 dimensional static acoustic MFCC features are extracted using a 25 ms analysis window with 10 ms shift. SDC features are added afterwards. In addition, voice activity detection is applied based on the log mel energy. We note that more advanced speech activity methods, such as Combo-SAD [28] are effective, but here we employ a more basic solution because data is generally noise free. A UBM with 1024 mixtures is trained on the given enrollment data. Specifically, the KALDI toolkit [29] is adopted for both acoustic feature extraction and UBM training which uses 20 iterations per mixture split. Based on the UBM, a total variability (TV) matrix is trained with the same enrollment data. Finally, 600 dimensional i-Vectors are extracted for each utterance.

Proposed Feature#1. Bottleneck based feature extraction (BNF#1): In this sub-section, we develop the first proposed feature which employs bottleneck framework. Since the unsupervised bottleneck feature (BNF#1) is extracted at the frame level, the corresponding proposed system is similar to our baseline, but using the bottleneck feature instead of acoustic features as input for the UBM and i-Vector extraction. However, autoencoders can be employed at two different levels. First, it is used for the discriminative feature extraction at the utterance level. Secondly, the autoencoders can generate advanced feature for each frame as an input acoustic feature.

Proposed Feature#2. Autoencoder based feature extraction (AEF#2): In this sub-section, we introduce the utterance-level feature extraction based on autoencoder. At the utterance level, the transformation based on autoencoder can be treated as an i-Vector framework, similarly. Based on a well trained UBM, there is an adapted GMM for each utterance. With the parameters of the adapted GMM (i.e., mean and variance) as input features to the autoencoder model, the latent value can be extracted as a discriminative feature at the utterance level (i.e., AEF#2). The autoencoder based proposed feature extraction diagram is shown in Fig. 4.

Proposed Feature#3. Autoencoder based feature extraction (AEF#3): In this sub-section, another utterance-level feature

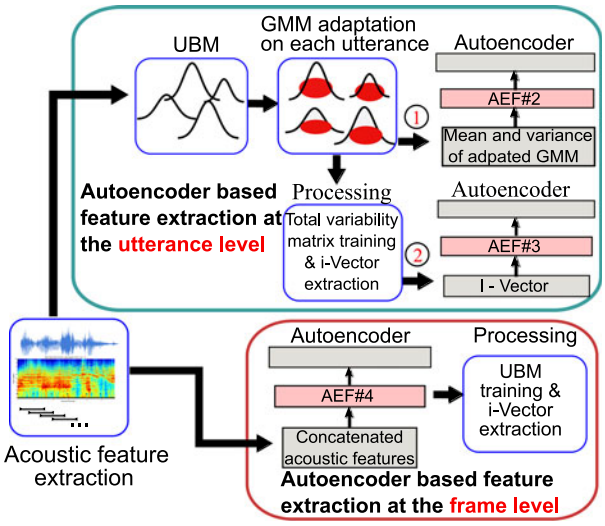

Fig. 4. Autoencoder based feature extraction diagram (explanations for the abbreviations can be found in Section IV).

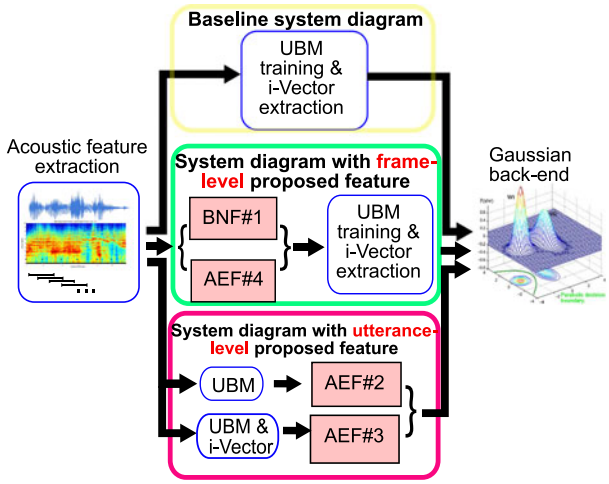

Fig. 5. Overall system diagram (explanations for the abbreviations can be found in Section IV).

extraction based on an autoencoder is proposed. We can assume that the latent value follows an arbitrary prior distribution. Instead of an adapted GMM parameter set, the i-Vector can be directly considered as input to the autoencoder. Therefore, the latent values are another autoencoder based proposed feature at the utterance-level (i.e., AEF#3).

Proposed Feature#4. Autoencoder based feature extraction (AEF#4): In this sub-section, a frame-level feature extraction based on autoencoder is proposed. Similar to the unsupervised bottleneck extraction, the autoencoder strategy can also be applied also at the frame level. With 11 frames concatenated filter bank features as input to the generative autoencoder model, the latent layer will output an advanced feature (i.e., AEF#4) for each frame. Subsequently, the proposed feature can be used for UBM training and i-Vector extraction which follows a Gaussian classifier back-end. The entire system diagram can be illustrated in Fig. 5.

Reference Feature. Traditional bottleneck feature (BNF): To compare with the feature extraction without additional transcribed corpus, traditional bottleneck (see Section II-A) are explored here. The corresponding framework similar to baseline, but using traditional bottleneck feature instead of acoustic features as input for the UBM and i-Vector extraction.

As noted in Section III, two types of autoencoders are employed in our study, which include the variational autoencoder and adversarial autoencoder. For simplicity, the proposed feature with suffix **-VA** represents that it is based on a variational autoencoder. Similarly, **-AA** means the extracted feature is based on an adversarial autoencoder. For example, AEF#4-VA and AEF#4-AA represent a frame-level proposed feature based on variational autoencoder and adversarial autoencoder, respectively.

TABLE I
CHINESE DIALECT

Dialect	CMN	HSN	WU	YU
Train (H)	6.3	8.9	5.1	7.7
Test (H)	2.2	2.9	1.7	2.6
Avg. Dur.		10 s		

V. EVALUATION CORPORA

The corpora utilized for evaluation in this study consists of a Chinese dialect dataset [25], Pan-Arabic corpus and multi-dialect multi-genre evaluation corpus (MGB-3) [30].

The Chinese corpus consists of four Chinese dialects (sub-languages): Mandarin (CMN), Cantonese (YU), Xiang (HSN), and Wu (WU). All data in this corpus is based on spontaneous conversational noise-free speech, without duration mismatch between training and testing data. More detailed information about training and test data is shown in Table I. General speaking, acoustic similarity among the target Chinese dialects is lower than regular dialects. For example, Mandarin speakers cannot understand Cantonese unless they are dedicated to learning this language.

The Pan-Arabic corpus consists of Arabic dialect data from five different regions, including United Arab Emirates (AE), Egypt (EGY), Iraq (IRQ), Palestine (PS), and Syria (SY). Each dialect set captures conversations of 100 speakers (gender balanced). To be consistent with the Chinese corpus statics, around 7 h and 2 h data per Arabic dialect are random selected from original corpus to make up training and test set, respectively. Our previous study [2] shows that Pan-Arabic is more challenging than Chinese because of linguistic similarity. Finally, with respect to consistent DID system evaluation, both corpora were collected in the countries/regions with the exact same recording system/equipment. A previous study by Boril and Hansen [31] "Is the secret in the silence?" showed that consistent recording conditions are needed for DID task.

Without loss of generality, MGB-3 is also used for DID evaluation. The speech data is broad and muti-genre, spanning the whole range of TV output. It include five major Arabic dialects: Egyptian (EGY), Levantine (LAV), Gulf (GLF), North African (NOR), and Modern Standard Arabic (MSA). In our study, the training set (around 10 h per dialect) is used for acoustic modelling and dev set (around 2 hours per dialect) are used for

system evaluation. In addition, MGB-2 data is encouraged for system development, which is approximate 1200 h of Arabic broadcast data from 4000 programmers broadcast on Aljazeera Arabic TV channel over a span of 10 years.

The public AMI meeting corpus with individual headset microphones is used for English acoustic modelling and traditional bottleneck feature extraction.

VI. Result and Analysis

This section focuses on the analysis of all LID/DID system performance using the proposed unsupervised learning methods. To evaluate the experiments in terms of different perspectives, two types of measurement criteria were adopted. The first is averaged accuracy across classes. In addition, an evaluation of the overall classification performance using the standard NIST LRE criterion average cost function (C_{avg}) [32] is employed,

$$C_{\text{avg}} = \frac{1}{N_L}\{[C_{\text{Miss}} \cdot P_{\text{Target}} \cdot P_{\text{Miss}}(L_T)] + \frac{1}{N_L - 1}[C_{\text{FA}} \cdot (1 - P_{\text{Target}}) \cdot P_{\text{FA}}(L_T, L_N)]\}, \tag{8}$$

where N_L is the number of languages in each language cluster, L_T and L_N are target and non-target languages. Other parameters are defined as $C_{\text{Miss}} = C_{\text{FA}} = 1$ and $P_{\text{Target}} = 0.5$.

A. Proposed Method Evaluation on Chinese Corpus

1) Impact of Unsupervised Bottleneck: To better visualize the impact of the proposed bottleneck feature with unsupervised phonetic labeling, a t-distributed stochastic neighbor embedding (t-SNE) [33] visualization process is adopted. This is a technique for dimensionality reduction that is particularly well suited for visualization of high-dimensional datasets. The visualized features are 600 dimensional i-Vectors which are extracted based on either acoustic feature MFCCs or our proposed bottleneck features, since it is the finalized input feature to the Gaussian classifier back-end. Here, t-SNE is set up with 200 iterations for both experiments. Fig. 6 shows that the distribution of i-Vectors from the baseline LID/DID system using MFCCs. It can be noted that the four dialects are basically separated, but there is some overlap between dialect WU and HSN, which is hard to distinguish perfectly. In contrast, Fig. 7 shows the i-Vectors based on the proposed new bottleneck feature. There is now a more clear boundary between each pair of dialects. Notably, the proposed feature is more discriminative than the acoustic MFCC features seen in i-Vectors from Fig. 6.

In terms of the impact on system performance, Table II shows evaluation results from two perspectives. It can be seen that traditional BNF (based on AMI) does not benefit the system performance compared to baseline. On the contrary, the accuracy decrease from 95.5% to 90.1%. One of the reason is that there are some severe mismatch between Chinese DID corpus and AMI corpus, such as, channel mismatch, speech format mismatch. With the proposed bottleneck feature, the overall cost performance $C_{\text{avg}} * 100$ deceases from 2.7 to 1.3 with a relative +58% improvement, and the average accuracy increases from 95.5% to 97.8%. This shows how effective the proposed solution is for a 4-way Chinese dialect task using BNF#1.

From the BNF#1 extracting diagram (Fig. 2), it can be noted DNN is mimic the behavior of GMM to predict the phonetic cluster label. Where do these benefits come from? Firstly of all, replacing GMM with DNN for speech acoustic modeling with significant improvement has already been proved in ASR, where the senone labels that used at the output layer are also generated

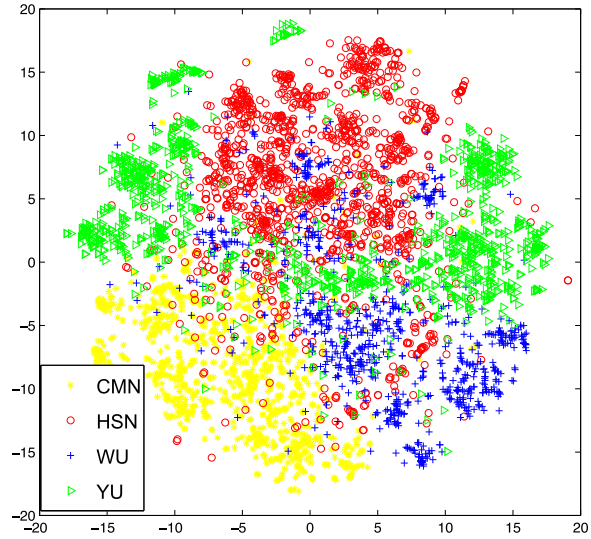

Fig. 6. Baseline system: i-Vector distribution visualization (t-SNE iter = 200).

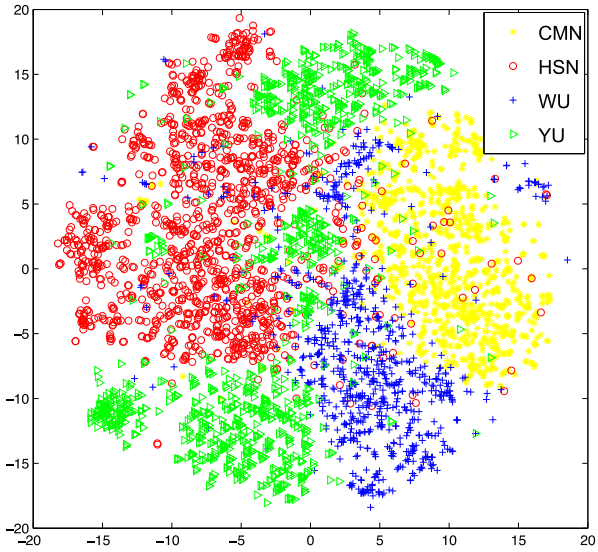

Fig. 7. BNF#1 based system: i-Vector distribution visualization (t-SNE iter = 200; explanations for the abbreviations can be found in Section IV).

TABLE II
Unsupervised Bottleneck Impact on Chinese (in %; Explanations for the Abbreviations Can be Found in Section IV)

	C_{avg}	Accuracy
Baseline (MFCC)	2.7	95.5
BNF (based on AMI)	5.4	90.1
BNF#1	**1.3**	**97.8**

TABLE III
AUTOENCODER IMPACT ON CHINESE DID (IN %; EXPLANATIONS FOR THE ABBREVIATIONS CAN BE FOUND IN SECTION IV)

		C_{avg}	Accuracy
	Baseline (MFCC)	2.7	95.5
AEF#2	AEF#2-VA	4.0	93.8
	AEF#2-AA	3.5	94.3
AEF#3	AEF#3-VA	5.8	90.8
	AEF#3-AA	6.3	89.9
AEF#4	AEF#4-VA	**2.5**	**95.7**
	AEF#4-AA	2.7	**95.7**

TABLE IV
SYSTEM FUSION ON CHINESE (IN %; EXPLANATIONS FOR THE ABBREVIATIONS CAN BE FOUND IN SECTION IV)

System Fusion	C_{avg}	Accuracy
Baseline + AEF#4-AA	1.9	96.9
Baseline + AEF#4-VA	2.0	97.1
Baseline + BNF#1	1.3	98.0
AEF#4-AA + AEF#4-VA + BNF#1	1.2	98.1
Baseline + AEF#4-AA + AEF#4-VA + BNF#1	1.2	98.1

through a traditional GMM-HMM. Similar strategy is utilized in our study. Secondly, the intermediate features which contains both acoustic and phonetic information can be extracted easily with this replacement, since the output layer is estimated phonetic label. Compared to MFCC, BNF#1 also contains phonetic information for each single frame, which contributes on further speech signal processing.

2) Impact of the Autoencoder: As noted in Section III, two types of autoencoders are employed in our study, which include the variational autoencoder and adversarial autoencoder. In addition, the autoencoder models can be utilized at three difference phases in the DID system according to Section IV (i.e., AEF#2, AEF#3, and AEF#4). Specifically, AEF#2 is based on the autoencoder which is employed at the utterance level with adapted GMM parameters. Similarly, the hidden layer of the autoencoder generates AEF#3 when the input feature is an i-Vector. In contrast, AEF#4 is extracted at the frame level, whose input is an 11 frame concatenated filter bank feature. Therefore, 2 * $3 = 6$ different feature sets based on the specific autoencoder position are proposed here in our study.

The performance of autoencoder based proposed three features applied to the Chinese dialect corpus are shown in Table III. At the utterance level, autoencoder models are better able to learn the discriminative information from the GMM parameters than from an i-Vector, since AEF#2 consistently outperforms AEF#3. However, neither of them is as completive as the baseline. Generally, it can be noted that the autoencoder model applied at the frame level works better than that at the utterance level. Specifically, C_{avg} decrease from 2.7% to 2.5% with a relative +7.4% improvement using the variational autoencoder applied at the frame-level. In addition, the adversiarial autoencoder at the frame level achieves slightly greater accuracy compared to our baseline system.

In general, the autoencoder is a generative model which needs adequate samples for the training phase. Notably, the autoencoder applied at the frame level will generate many more samples than applied at the utterance level. This might be an essential reason why the frame-level proposed feature works much better than at the utterance-level. In addition, since an i-Vector is a very discriminative feature, the autoencoder model is less effective in improving information extraction. Therefore, only the frame-level AEF#4 is considered for further evaluation.

Frame-level variational/adversarial autoencoder imposes an arbitrary distribution on the observed data (i.e., 11 concatenated frame acoustic feature). Specifically, the feature uses KL divergence and adversarial training procedure to encourage a latent code distribution to match an arbitrary one, respectively. According to DNN-based acoustic modeling, the bottleneck layer contains phonetic information since the senone labels are assigned to the output layer. Similarly, the latent code vector presents some phonetic information as well, since it can be used to represent/reconstruct the observed data. Therefore, all the proposed methods applied at the acoustic feature level bring benefit, because more phonetic information is embedded in the resulting proposed feature.

3) Impact of System Fusion: The complementarity of proposed features are explored in this section by fusing the system scores with FoCal Toolkit [34]. Table IV shows that the fusion of autoencoder based features (AEF#4-VA, AEF#4-AA) and baseline (MFCC) bring around 29% improvement on C_{avg}. Compared to BNF#1, autoencoder based features do not show many benefits on single system. However, they are complementary with baseline so that the fusion improve the performance significantly. Using all the proposed features can achieve the best performance with 98.1% accuracy.

4) Noise Robustness Analysis of Effective Proposed Methods: The unsupervised bottleneck feature (BNF#1) and frame-level applied autoencoder (AEF#4) have already been shown to be effective on clean data. However, noisy conditions are a challenge in real scenarios. To simulate noisy speech signals, speech-shape noise (SSN) is employed in our study whose long-term average spectrum is similar to that of speech. In order to evaluate the robustness of our proposed methods under controlled noisy conditions, the performance with different speech-to-noise-ratios (SNR) is illustrated in Fig. 8. It can be noted that the unsupervised bottleneck feature outperforms the baseline MFCCs on noisy speech with SNR from 5 to 20 dB. In particular, the unsupervised bottleneck feature achieves a relative +34% improvement compared to baseline at the 15 dB SNR.

From statistic perspective, noisy condition may directly change the mean and variance of each phonetic cluster, but not impact significantly on the distribution of phonetic space. For single frame of speech, the estimated phonetic label might not be interrupted as much as the extracted acoustic feature. Since bottleneck layer is close to output layer, BNF#1 becomes more robust towards estimated phonetic labels. In other words, the impact of noisy conditions is attenuated on bottleneck layer, because of the deep neural network structure. However, the variational/adversarial autoencoder doesn't benefit on noisy robustness. Compared with DNN-based acoustic modeling (4-layers), the autoencoder is a shallow network which is less robust to noise corruption.

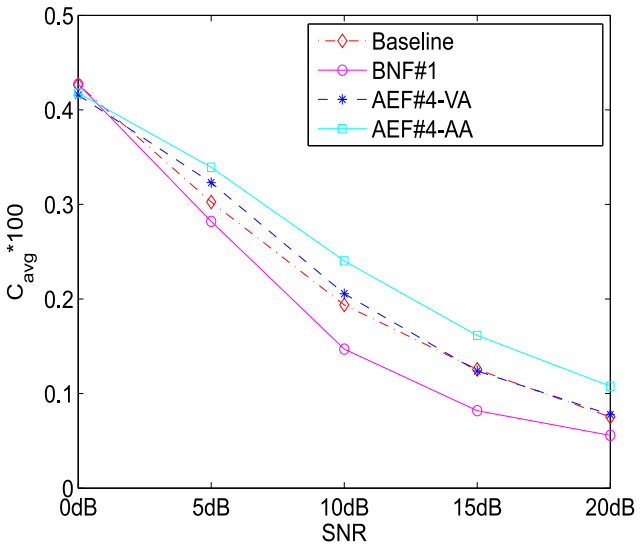

Fig. 8. Noise robustness analysis on Chinese ($C_{\text{avg}} * 100$; explanations for the abbreviations can be found in Section IV).

TABLE V
UNSUPERVISED BOTTLENECK IMPACT ON PAN-ARABIC (IN %; EXPLANATIONS FOR THE ABBREVIATIONS CAN BE FOUND IN SECTION IV)

		C_{avg}	Accuracy
Baseline (MFCC)		16.2	72.0
BNF (based on AMI)		13.2	78.7
BNF#1		**12.8**	**81.3**
AEF#4	AEF#4-VA	**14.6**	74.8
	AEF#4-AA	15.4	**74.9**

B. Proposed Methods Evaluation on the Pan-Arabic Corpus

1) Impact of Proposed Methods: Additional evaluations on a closely space DID task are explored in this section in order to demonstrate the effectiveness of the proposed methods. The Pan-Arabic corpus is considered as an additional evaluation dataset which is more challenging compared to the Chinese corpus. Similarly, three effective methods (BNF#1, AEF#4-VA and AEF#4-AA) are adopted here, since they have already shown promising improvement on the Chinese DID task.

Table V shows that all proposed features outperform the i-Vector baseline using MFCCs for the 5-way Arabic DID. Specifically, the proposed bottleneck solution achieves 81.3% on accuracy with a relative +12.9% improvement. With tradition BNF trained with AMI, the performance is better than baseline, but still not competitive with proposed BNF#1. In addition, adversarial autoencoder achieves slightly better performance than the variational autoencoder in terms of accuracy. More numerical details can be found in Table V. Compared with performance on the Chinese corpus, it can be seen that the adversarial autoencoder has greater benefit when the language/dialects are closer to each other. In addition, the unsupervised bottleneck feature achieves the best performance across both Chinese and Pan-Arabic corpora.

2) Impact of System Fusion: Arabic system fusion is investigated in this section. Similar to Chinese DID, each proposed feature is complementary to MFCC, so that the system fusion

TABLE VI
SYSTEM FUSION ON PAN-ARABIC (IN %; EXPLANATIONS FOR THE ABBREVIATIONS CAN BE FOUND IN SECTION IV)

System Fusion	C_{avg}	Accuracy
Baseline + AEF#4-AA	12.6	78.5
Baseline + AEF#4-VA	11.9	78.9
Baseline + BNF#1	10.6	81.3
AEF#4-AA + AEF#4-VA + BNF#1	9.3	83.6
Baseline + AEF#4-AA + AEF#4-VA + BNF#1	8.9	84.7

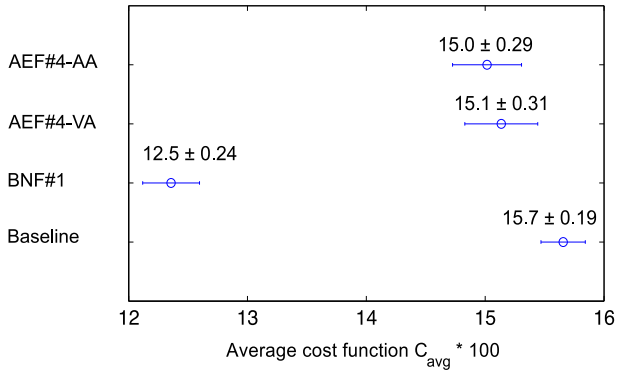

Fig. 9. Performance robustness analysis on Arabic (C_{avg} * 100).

of two features (one proposed feature and MFCC) improves the system performance consistently (see Table VI). By fusing the scores of proposed systems (i.e., BNF#1, AEF#4-VA and AEF#4-AA), we achieve 9.3 on C_{avg} and 83.6% on accuracy. Furthermore, the fusion of all proposed features and MFCC achieves the best system performance. Compared to baseline, the best fusion system achieves 8.9 on C_{avg} with a relative 45.1% improvement.

3) Performance Robustness Analysis: LID/DID system performance can be hindered by many speech characteristics, such as a gender imbalance, age mismatch, speaker variation, etc.(i.e., similar to SID performance as discussed in [15]). Also, the "closeness" of dialects has a major impact on final classification performance. The study by Mehrabani and Hansen [3] explored automatic ways to assess dialect separation as means of understanding how close individual dialects might be for a 3-way DID task. In this section, a performance analysis on robustness is conducted through evaluating the system on 6 different test sets, which might contain different speakers with age and gender mismatch. Fig. 9 shows the C_{avg} mean and variance for Arabic DID based on the three proposed methods as well as baseline. Notably, three proposed features (i.e., BNF#1, AEF#4-VA and AEF#4-AA) achieve lower mean but higher variance compare to the baseline solution. Compared to variational autoencoder (AEF#4-VA), adversarial autoencoder (AEF#4-AA) achieves greater benefit when the languages/dialects are closer. Therefore, it can be noted that the adversarial training procedure makes the latent variable slightly more sensitive to alternate speech characteristics.

C. Proposed BNF#1 Evaluation on MGB-3

Without loss of generality, the most effective proposed method BNF#1 is also evaluated on MGB-3 challenge. Here,

TABLE VII
SYSTEM PERFORMANCE ON MGB-3 (IN %; EXPLANATIONS FOR THE ABBREVIATIONS CAN BE FOUND IN SECTION IV)

System	Accuracy
MFCC	51.2
BNF* (based on Arabic corpus)	57.8
BNF#1	57.0
MFCC (with MGB-2 data)	61.6
BNF#1 (with MGB-2 data)	65.4

1024 Gaussian mixtures are set for UBM training and 2048 Gaussian mixtures are used for universal phonetic space modeling. In addition, bottleneck features (BNF*) extracted from an DNN-based Arabic ASR is adopted here. As [30] mentioned, the system is based on two successive DNN models. Both DNNs use the same setup of 5 hidden sigmoid layers and 1 linear bottleneck layer [35]. Specifically, 60 h of manually transcribed Al-Jazeera MSA news recordings is used for DNN training [36]. For comparison, the BNF* based i-Vectors for MGB-3 data is evaluated in our study. Table VII shows that i-Vector system based on BNF* is slightly better than proposed BNF#1. Furthermore, using more MGB-2 data for UBM and TV Matrix training, the system performance is boosted to 65% by using proposed BNF#1. Without any additional manually transcribed Arabic corpus, the proposed method can improve Arabic DID from 51.2% to 65.4%.

VII. CONCLUSION

Traditional bottleneck feature extraction with an i-Vector framework has been used for state-of-the-art language/dialect identification (LID/DID). However, this approach has a major limitation in that it requires additional outside transcribed speech information for ASR acoustic modeling. In this study, two types of unsupervised deep learning methods have been introduced. First, an unsupervised bottleneck feature extraction solution was proposed, which was derived from a traditional structure but trained with estimated phonetic labels requiring no secondary transcribed data. In addition, two types of latent variable learning algorithms based on generative autoencoder model were introduced for speech feature processing, which were applied at three separate phases. To demonstrate the effectiveness of the proposed methods, three dialect corpora were evaluated in our study. It can be noted that the proposed unsupervised bottleneck feature (BNF#1) achieves the best performance across three corpora. Specifically, it achieves a relative relative +58% improvement on C_{avg} for a 4-way Chinese dialect corpus. Even under noisy conditions, the unsupervised bottleneck solution consistently outperforms MFCCs. Additionally, it was shown that the proposed autoencoder applied at the frame-level (AEF#4) works better than that applied at the utterance level (AEF#2 and AEF#3), since more phonetic information are embedded to extracted the latent variables. Specifically, the variational autoencoder outperforms the adversarial autoencoder for Chinese DID. Also, the adversarial autoencoder achieve greater gains more than the variational autoencoder for the 5-way Pan-Arabic DID task. It can be noted that the adversarial autoencoder is more beneficial when the dialects are close to each other.

The resulting solutions here show how improved feature and feature extraction methods, based on a bottleneck framework and an autoencoder placed at different locations, improves discrimination for the challenging research problem of closely spaced dialects in a DID task.

REFERENCES

[1] D. Martinez, L. Burget, L. Ferrer, and N. Scheffer, "iVector-based prosodic system for language identification," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, Kyoto, Japan, Mar. 2012, pp. 4861–4864.

[2] Q. Zhang, H. Boril, and J. H. L. Hansen, "Supervector pre-processing for PRSVM-based Chinese and Arabic dialect identification," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, Vancouver, BC, Canada, May 2013, pp. 7363–7367.

[3] M. Mehrabani and J. H. L. Hansen, "Automatic dialect separation assessment," *Int. J. Speech Technol.*, vol. 18, pp. 277–286, 2015.

[4] L. Mary and B. Yegnanarayana, "Extraction and representation of prosodic features for language and speaker recognition," *Speech Commun.*, vol. 50, no. 10, pp. 782–796, 2008.

[5] D. Garcia-Romero and A. McCree, "Supervised domain adaptation for i-vector based speaker recognition," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, Florence, Italy, May 2014, pp. 4047–4051.

[6] G. Liu and J. H. L. Hansen, "An investigation into back-end advancements for speaker recognition in multi-session and noisy enrollment scenarios," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 22, no. 12, pp. 1978–1992, Dec. 2014.

[7] D.-Cheng Lyu, R.-Yuan L., Y.-C. Chiang, and C.-N. Hsu, "Speech recognition on code-switching among the chinese dialects," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. Proc.*, 2006, vol. 1, pp. 1105–1108.

[8] Q. Zhang and J. H. L. Hansen, "Training candidate selection for effective rejection in open-set language identification," in *Proc. SLT: Spoken Lang. Technol. Workshop*, South Lake Tahoe, NV, USA, 2014, pp. 384–389.

[9] M. F. BenZeghiba, J. Gauvain, and L. Lamel, "Gaussian backend design for open-set language detection," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, Taipei, Taiwan, 2009, pp. 4349–4352.

[10] P. A. Torres-Carrasquillo, D. A. Reynolds, and J. R. Deller Jr., "Language identification using Gaussian mixture model tokenization," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, Orlando, FL, USA, May 2002, pp. 757–760.

[11] P. Kenny, G. Boulianne, P. Ouellet, and P. Dumouchel, "Joint factor analysis versus eigenchannels in speaker recognition," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 15, no. 4, pp. 1435–1447, May 2007.

[12] D. Martınez, O. Plchot, L. Burget, O. Glembek, and P. Matejka, "Language recognition in iVectors space," in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, Florence, Italy, Sep. 2011, pp. 861–864.

[13] W. Shen, W. Campbell, T. Gleason, D. Reynolds, and E. Singer, "Experiments with lattice-based PPRLM language identification," in *Proc. Odyssey: Speaker Lang. Recognit. Workshop*, Puerto Rico, 2006, pp. 1–6.

[14] M. N., S. Safavi, P. Weber, and M. Russell, "Identification of British English regional accents using fusion of i-vector and multi-accent phonotactic systems," in *Proc. ODYSSEY: Speaker Lang. Recognit. Workshop*, 2016, pp. 213–218.

[15] J. H. L. Hansen and T. Hasan, "Speaker recognition by machines and humans: A tutorial review," *IEEE Signal Process. Mag.*, vol. 32, no. 6, pp. 74–99, Nov. 2015.

[16] Y. Lei, N. Scheffer, L. Ferrer, and M. McLaren, "A novel scheme for speaker recognition using a phonetically-aware deep neural network," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, Florence, Italy, May 2014, pp. 1695–1699.

[17] P. Kenny, V. Gupta, T. Stafylakis, P. Ouellet, and J. Alam, "Deep neural networks for extracting baum-welch statistics for speaker recognition," in *Proc. Odyssey: Speaker Lang. Recognit. Workshop*, Joensuu, Finland, Jun. 2014, pp. 293–298.

[18] Q. Zhang, G. Liu, and J. H. L. Hansen, "Robust langauge recognition based on diverse feature," in *Proc. Odyssey: Speaker Lang. Recognit. Workshop*, Joensuu, Finland, Jun. 2014, pp. 152–l57.

[19] P. Matejka *et al.*, "Neural network bottleneck features for language identification," in *Proc. Odyssey: Speaker Lang. Recognit. Workshop*, Joensuu, Finland, Jun. 2014, pp. 299–304.

[20] Y. Zhang, E. Chuangsuwanich, and J. Glass, "Language id-based training of multilingual stacked bottleneck features," in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, 2014, pp. 1–5.
[21] K. A. Lee *et al.*, "The 2015 NIST language recognition evaluation: The shared view of i2r, fantastic4 and singams," in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, 2016, pp. 3211–3215.
[22] G. E. Dahl, D. Yu, L. Deng, and A. Acero, "Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition," *IEEE Trans. Audio, Speech, Lang. Process*, vol. 20, no. 1, pp. 30–42, Jan. 2012.
[23] D. P. Kingma and M. Welling, "Stochastic gradient vb and the variational auto-encoder," in *Sec. Int. Conf. Lear. Rep., ICLR Talk Slides*, 2014.
[24] A. Makhzani, J. Shlens, N. Jaitly, and I. J. Goodfellow, "Adversarial autoencoders," arXiv preprint arXiv:1511.05644, 2015.
[25] Y. Lei and J. H. L. Hansen, "Dialect classification via text-independent training and testing for Arabic, Spanish, and Chinese," *IEEE Trans. Audio, Speech, Lang. Process*, vol. 19, no. 1, pp. 85–96, Jan. 2011.
[26] S. J. Young, J. J Odell, and P. C. Woodland, "Tree-based state tying for high accuracy acoustic modelling," in *Proc. Workshop Human Lang. Technol.*, 1994, pp. 307–312.
[27] M. McLaren, L. Ferrer, and A. Lawson, "Exploring the role of phonetic bottleneck features for speaker and language recognition," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, Shanghai, China, Mar. 2016, pp. 5575–5579.
[28] S. O. Sadjadi and J. H. L. Hansen, "Unsupervised speech activity detection using voicing measures and perceptual spectral flux," *IEEE Signal Process Lett.*, vol. 20, no. 3, pp. 197–200, Mar. 2013.
[29] D Povey *et al.*, "The kaldi speech recognition toolkit," in *Proc. ASRU: IEEE Workshop Autom. Speech Recognit. Underst.*, Waikoloa, HI, USA, Dec. 2011.
[30] A. Ali *et al.*, "Automatic dialect detection in arabic broadcast speech," in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, San Francisco, CA, USA, 2015, pp. 2934–2938.
[31] H. Bořil, A. Sangwan, and J H. L. Hansen, "Arabic dialect identification - 'Is the secret in the silence?' and other observations," in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, Portland, OR, USA, Sep. 2012, pp. 30–33.
[32] Alvin Martin and Craig Greenberg, "The 2009 NIST language recognition evaluation," in *Proc. Odyssey: Speaker Lang. Recognit. Workshop*, Brno, Czech Republic, Jun. 2010, pp. 165–171.
[33] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, pp. 2579–2605, 2008.
[34] N. Brümmer, "Focal multi-classtools for evaluation, calibration and fusion of, and decision-making with, multi-class statistical pattern recognition scores," 2007. [Online]. Available: http://sites. google. com/site/nikobrummer/focalmulticlass
[35] P. Cardinal, N. Dehak, Y. Zhang, and J. R. Glass, "Speaker adaptation using the i-vector technique for bottleneck features," in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, 2015, pp. 2867–2871.
[36] A. Ali, Y. Zhang, P. Cardinal, N. Dahak, S. Vogel, and J. Glass, "A complete kaldi recipe for building arabic speech recognition systems," in *Proc. IEEE Spoken Lang. Technol. Workshop*, 2014, pp. 525–529.

Qian Zhang received the B.S. degree in electrical engineering from Zhengzhou University, Zhengzhou, China, in 2011. She started working toward the Ph.D. degree at the University of Texas at Dallas (UTD), Richardson, TX, USA, at same year. Since 2012, she has been a Graduate Research Assistant with the Center for Robust Speech Systems (CRSS), UTD. She served as the Lead CRSS-UTD student in the NIST Language Recognition Evaluation challenge 2015 and also participated several other challenges on speech area, such as NIST 2015 Language Recognition I-vector Machine Learning Challenge, Automatic Speaker Verification Spoofing 2015 Challenge, and MGB Arabic Dialect Identification 2017 Challenge. Her research interests include data selection and enhancement for open-set language identification, diverse feature extraction, unsupervised deep learning used for acoustic modeling. She has authored/co-authored around ten journal and conference papers in the field of speech processing and language technology. She was a recipient of the IBM Best Student Paper Award at IEEE ICASSP 2016.

John H. L. Hansen (S'81–M'82–SM'93–F'07) received the B.S.E.E. degree from the College of Engineering, Rutgers University, New Brunswick, NJ, USA, in 1982, the M.S. and Ph.D. degrees in electrical engineering from Georgia Institute of Technology, Atlanta, GA, USA, in 1983 and 1988, respectively.

He joined Erik Jonsson School of Engineering and Computer Science, University of Texas at Dallas (UTDallas), in 2005, where he currently serves as a Jonsson School Associate Dean for Research, as well as a Professor of Electrical and Computer Engineering, the Distinguished University Chair in Telecommunications Engineering, and a joint appointment as a Professor in the School of Behavioral and Brain Sciences (Speech & Hearing). He previously served as a UTDallas Department Head of Electrical Engineering from August 2005 to December 2012, overseeing a +4× increase in research expenditures (4.5 to 22.3 M) with a 20% increase in enrollment along with hiring 18 additional T/TT faculty, growing UTDallas to the eighth largest EE program from ASEE rankings in terms of degrees awarded. At UTDallas, he established the Center for Robust Speech Systems (CRSS). Previously, he served as a Department Chairman and Professor of Speech, Language and Hearing Sciences, and a Professor in Electrical and Computer Engineering, University of Colorado-Boulder (1998–2005), where he co-founded and served as an Associate Director of the Center for Spoken Language Research. In 1988, he established the Robust Speech Processing Laboratory and continued to direct research activities in CRSS at UTDallas. He is the author/coauthor of 678 journal and conference papers including 12 textbooks in the field of speech processing and language technology, signal processing for vehicle systems, coauthor of textbook: *Discrete-Time Processing of Speech Signals* (IEEE Press, 2000), co-editor of DSP for *In-Vehicle and Mobile Systems* (Springer, 2004), *Advances for In-Vehicle and Mobile Systems: Challenges for International Standards* (Springer, 2006), *In-Vehicle Corpus and Signal Processing for Driver Behavior* (Springer, 2008), and lead author of the report The Impact of Speech Under Stress on Military Speech Technology, (NATO RTO-TR-10, 2000). He has supervised 85 Ph.D./M.S. thesis candidates (47 Ph.D., 38 M.S./M.A.). His research interests include the areas of digital speech processing, analysis and modeling of speech and speaker traits, speech enhancement, feature estimation in noise, signal processing for hearing impaired/cochlear implants, robust speech recognition with emphasis on machine learning and knowledge extraction, and in-vehicle interactive systems for hands-free human-computer interaction. Dr. Hansen received the honorary degree Doctor Technices Honoris Causa from Aalborg University (Aalborg, DK) in April 2016, in recognition of his contributions to speech signal processing and speech/language/hearing sciences. He was recognized as an IEEE Fellow (2007) for contributions in "Robust Speech Recognition in Stress and Noise," International Speech Communication Association (ISCA) Fellow (2010) for contributions on research for speech processing of signals under adverse conditions, and received The Acoustical Society of Americas 25 Year Award (2010) in recognition of his service, contributions, and membership to the Acoustical Society of America. He is currently serving as ISCA President (2017–2019) and a member of the ISCA Board, having previously served as the Vice-President (2015–2017). He also is serving as a Vice-Chair on U.S. Office of Scientific Advisory Committees (OSAC) for OSAC-Speaker in the voice forensics domain (2015–2017). Previously, he served as an IEEE Technical Committee (TC) Chair and member of the IEEE Signal Processing Society: Speech-Language Processing Technical Committee (SLTC) (2005–2008; 2010–2014; elected IEEE SLTC Chairman for 2011–2013, Past-Chair for 2014), and elected as an ISCA Distinguished Lecturer (2011–2012). He has served as the member of the IEEE Signal Processing Society Educational Technical Committee (2005–2008; 2008–2010); Technical Advisor to the U.S. Delegate for NATO (IST/TG-01); IEEE Signal Processing Society Distinguished Lecturer (2005–2006), Associate Editor for IEEE TRANSACTION SPEECH AND AUDIO PROCESSING (1992–1999), Associate Editor for IEEE SIGNAL PROCESSING LETTERS (1998–2000), Editorial Board Member for IEEE SIGNAL PROCESSING MAGAZINE (2001–2003); and Guest Editor (October 1994) for special issue on Robust Speech Recognition for IEEE TRANSACTION SPEECH AND AUDIO PROCESSING. He is serving as an Associate Editor for JASA, and served on Speech Communications Technical Committee for Acoustical Society of America (2000–2003). He was the recipient of The 2005 University of Colorado Teacher Recognition Award as voted on by the student body. He organized and served as the General Chair for ISCA Interspeech-2002, September 16–20, 2002, Co-Organizer, and Technical Program Chair for IEEE ICASSP-2010, Dallas, TX, USA, March 15–19, 2010, and Co-Chair and Organizer for IEEE SLT-2014, December 7–10, 2014 in Lake Tahoe, NV, USA.