Investigation on how to improve the performance of LID system under low-resource and multi-domain conditions.

Abstract

low-resource and multi-domain are enormous challenges of the spoken language recognition(LID), but there are seldom studies on this issue so far. Thus this paper focuses on investigate how to improve the performance of LID system under low-resource and multi-domain condition. This paper applies different systems including traditional iVector system and xVector system used in speaker recognition. The acoustic representations MFCC and Fbank. Additionally, the bottleneck feature, different backend scoring and data augmentation are also applied in this paper. Moreover, we are first to utilize Multi-language BN and domain adaptation based on the distribution of in-domain and out-of-domain data in this task. In the post-evaluation analysis presented here, we try several variations of the language system, and find that the best performing system is bottleneck features with domain adaptation and Logistic regression(LR) classifier.

Index: low-resource, Multi-domain, Multi-language BN, Domain adaptation, LID

1. Introduction

Spoken language identification(LID) is expanding the broad range of applications to automatically determine which language is the given utterance. Generally speaking, there are two main LID categories: the acoustic and phonotactics approaches [1, 2, 3, 4]. The acoustic approach extracts discriminative features from the speech signals and build models with these features, such as Gaussian mixture models(GMMs) [5], to determine the language identity. The phonotactics approach is usually accomplished by phone recognition followed by language modeling (PRLM) [6]. Firstly, decoding the utterance into a sequence of phones, then an interpolated n-gram language model is used to estimate the probability of the obtained phone streams. The probability is often different for different languages to be identified. It can also be extended to Parallel-PRLM (PPRLM) which is incorporating multiple language dependent phone recognizers and building the corresponding set of language models [3]. LID typically focuses on a single domain during evaluation of unseen data. However, some languages are low-resource and it is hard to collect enough data on the same channel. This problem motivates the development of novel approaches that the knowledge distilled from the source domain can be transferred to the target domain.

In this paper, we investigate on how to improve the performance of LID system under low-resource and multi-domain condition. The systems rely on five types of vectors to represent speech from audio recording: mfcc-iVectors, fbank-xVectors, BNFs-xVectors, Multi-languages BNFs xVectors and domain adaptation xVectors. We find out two main advantages with Multi-language BNFs. First, these units are defined universally across multiple languages [4,7,8]. As a result, it alleviates the problem missing phones in the front-end phone recognizer of PRLM systems [3]. It facilitates sharing of speech data from different languages to enhance the modeling capability. Second, different acoustic definitions often exhibit complementary discrimination power. For each individual system, different acoustic features and models are adopted, and the overall performance is often additive when they are combined. The advantages of domain adaptation are especially prominent in out-of-domain data sets, it is discriminative for the main learning task on the source domain and can adapt to different channels.

In Session 2, we first describe the training and test data setup followed by a description of the two methods to solve the low-resource and multi-domain problem: Multi-language BNFs and

domain adaptation; In Section 5, the experimental setup is described; In Section 6, the results is represented; And in Section 7, the conclusions are derived.

2. Corpora

2.1 Training data

The experimental training set corpus are taken from the 2018 Oriental Language Recognition Competition, which was jointly organized by Tsinghua University and Haitian Ruisheng Company [9]. The corpus is provided by Haitian Ruisheng and contains 10 languages. The channel source of the voice is the traditional telephone channel with a frequency of 16 kHz. Each language is about 10 hours, and the gender ratio of men and women is 1:1. In order to emphasize the influence of data on language recognition, this experiment divides the data set into training data of four data equilibrium quantity sets, which are 25h, 50h, 75h, 106h, and are named respectively train_25h train_50h, train_75h, train_106h.

2.2 Evaluation

Our evaluation consists of two distinct datasets: in-domain test sets and out-of-domain test sets. The in-domain database is the standard test set for AP18-OLR, which contains the same target 10 languages, containing 1800 utterances each. The signals are recorded by mobile phones with a sampling rate of 16 kHz and a sample size of 16 bits. The out-of-domain test sets are downloaded from the Internet，which contains the target 6 languages, containing about 1800 utterances each. Before extracting the features of the out-of-domain test speech segment, the parameters of the speech are normalized to a sample rate of 16 kHz, a sample size of 16 bits and saved in wav format. ~~The training set and the in-domain and out-of-domain test sets are divided as Table 1.~~ The total duration of the training data is 106.58h, the total duration of the in-domain data in the set is 34.05h, and the total duration of the out-of-domain data is 15.71h.

3. Multi-language BNFs

In this paper, we explore fusion of multiple systems with different speech units due to the advantage and complementarity of universal speech attributes to language-dependent phonemes. The accuracy of phone recognizer is critical, but not the only factor for LID performance in the phonotactic approaches. In other words, it is fine to model the phonemes in the language model based on the assumption of similarity between these two language if a phoneme of another language to be recognized is always recognized as the one in the phone set designed for the phone recognizer. It is quite common for spoken languages in different language families that the phonemes cannot be represented well in language modeling if some phonemes are very different from the language for phone recognizer.

We could relieve this problem by using attribute units that are potentially language-universal across all spoken languages. Meanwhile, a single LID system may not achieve the comparable performance of a PR based LID system [10]. Because the size of the attribute inventory for manner or place of articulation is small. In this study, we show the complementary nature of speech attribute detectors to phone recognizers by fusing multiple language BNFs with phones and attributes.

4. Domain adaptation

The domain adaptation(DA) is to solve the target domain data insufficient (and possibly unlabeled) problem while source domain data(assumed labeled and sufficient for training a model) should be leveraged as well for training a model from scratch. Despite the differences in the marginal

distributions of the two domain, the knowledge distilled from the source domain can be transferred to the target domain. Due to the target data being weakly-labeled or even unlabeled, conventional approaches such as fine-tuning a source domain model to the target domain data may fail in many settings

Researcher pay more attention to DA method. Because many real-word applications large amounts of target domain labeled data are rarely available. Hence, for training the new models which require several thousands of training utterances, resorting to large out-of-domain corpora and using the small and possibly unlabeled target domain datasets for channel or other types of adaptation is necessary.

In this paper, we study the use of the DA, which approach the problem as a transformation of fixed utterance-level representations xVectors. We evaluate this method on the challenging task of channel adaptation. Our target is to greatly improve the accuracy of the out-of-doamin test data when the accuracy in in-domain test data is not greatly reduced. Finally, we utilize large amounts of source training data to training xVector extractor, and utilize very small amount annotate out-of-domain data to get xVectors of each language.

5. Experiment setup

We build several systems to investigate how to improve the performance of LID system under low-resource and multi-domain condition. All systems are built using the Kaldi speech recognition toolkit [11].

5.1 Baseline

iVector: Our acoustic-feature baseline system is a traditional iVector system. This system is based on the GMM-UBM recipe described in [12]. The features are 13 MFCCs with a frame-length of 25ms. They are mean normalized over a sliding window of up to 3 seconds. Delta and acceleration are appended to create 39 dimension feature vectors. An energy-based speech activity detection (VAD) system selects features corresponding to speech frames. The UBM is a 1024 component full-covariance GMM. The system uses a 400 dimensional iVector extractor.

xVector: The xVector system is based on a framework that developed for speaker recognition [13]. The recipe is based on the SRE16 v2 recipe available in the main branch of Kaldi as https://github.com/kaldi-asr/kaldi/tree/master/egs/sre16/v2. The feature learning component is a 5-layer time-delay neural network (TDNN). The statistic pooling layer computes the mean and standard deviation of the frame-level features from a speech segment. The size of the output layer is 10, corresponding to the number of languages in the training set. Once trained, the 512-dimensional activations of the penultimate hidden layer are read out as an xVector.

5.2 features

5.2.1 Acoustic features

The acoustic features are 39 mfccs with a frame-length of 25ms in iVector systems and 40 fbanks in xVector system.

5.2.2 English BNFs

It is difficult to extract the latent language information because of the diverse variations in speech utterances caused by different speakers, channels and banckground noise. It is most important to find effective representations of language information.

The language models are trained by alignments provided by a standard chain ASR model. 1300h of training data is used, and its input features are 40 fbanks. The dnn has 11 layers, and its

total left-context is 21 and right-context is 21. The softmax output layer computes posteriors for 5297 triphone states. Excluding the output layer, the dnn has 19.96 million parameters.

5.2.3 Multi-language BNFs

For the multi-domain condition, we use 256-dimensional BNFs extracted from an ASR DNN trained on multiple languages. Including the bottleneck layer, the DNN has the same architecture as Section 5.2.2. It also uses the same features.

The DNN is trained on 2 languages (Chinese and English). Among them, English ASR model is same with Section 5.2.2. Chinese ASR model is trained with 3000h training data, which architecture is same with English ASR model except that the posteriors is 5984 triphone states.

5.3 Classifer

Cosine: Simple cosine distance

Pos: Direct classification of xVector

Lr : Logistic regression

L-PLDA: LDA-based projection (9-dim in in-domain and <=9-dim in out-of-domain) plus PLDA scoring.

5.4 Data augmentation

The data augmentation methods to increase the amount and diversity of the iVector training data and the xVector DNN training data are as follows: speed perturbation, volume perturbation, reverberation and additive noises.

Speed perturbation is using a specified speed factor[14] to change the speed of the speech signal. Reverberation is convolving room impulse responses (RIR) with audio. For additive noise, we use the MUSAN dataset, which consists of over 900 noises, 42 hours of music from various genres and 60 hours of speech from twelve languages[15]. Both MUSAN and the RIR datasets are from http://www.openslr.org.

Finally, we use two ways of data augmentation, one is superimposed, which consists of 2-fold augmentation that combines the original "clear" training data with 1 mixed noise of multiple noises. The other is combined, which consists of 5-fold augmentation that combines the original "clean" training data with 4 copies of augmented data. To augment a recording, we randomly choose between one of the following:

- Speed perturbation: apply 1.1 times or 0.9 times speed of the original recording.
- Volume perturbation: the volume of the recording be chosen randomly to be between scale-low=0.125 and scale-high=2.
- Reverberation: the artficially reverberated data is convoluted with simulated RIRs.
- Babble: adding the summation of the speech from several speakers randomly selected from MUSAN[16] to the to the original signal (13-20dB SNR)
- Music: adding a randomly selected music file from MUSAN to the original signal (5-15dB SNR).
- Noise: adding MUSAN noises at one second intervals throughout the recording (0-15dB SNR).

6. Result

The evaluation standard is the accuracy metric. In the following tables, scores from each data source (in-domain or out-of-domain) or language have been balanced and contribute equally to the metric.

6.1 Baseline

In this section, we compare the performances of two state-of-the-art joint iVector systems and the xVector system in different duration of training data and different channels of test data.

MFCC is the input feature of the system iVec_mfcc_lr. Fbank is the Fbank of the xVector_fbank_lr. And their back-end is LR.

system	in-domain				Out-of-domain			
	25h	50h	75h	106h	25h	50h	75h	106h
iVec_mfcc_lr	71.43	84.00	88.05	90.62	31.94	37.28	40.15	37.51
xVec_fbank_lr	61.70	76.76	82.87	86.34	31.05	35.56	37.76	35.48

Table 1: comparing the accuracy of different durations of training sets in in-domain and out-of-domain. All systems conform to the fixed training condition.

In Table 1, we find that the smaller the amount of training data, the lower the accuracy in the in-domain. And the accuracy of out-of-domain is much lower than in-domain on same training data. Overall, in xVec_fbank_lr, the accuracy of in-domain in train_106h is 64.04% which is better than out-of-domain in train_25h. The experimental results above demonstrate the influence of low-resource and cross-channel on the accuracy of language recognition.

6.2 Classification Analysis

In this paper, we use four back-ends classifer as session 5.3.

system	in-domain				Out-of-domain			
	25h	50h	75h	106h	25h	50h	75h	106h
xVec_fbank_pos	58.00	72.77	78.92	81.66	27.98	30.07	28.60	28.73
xVec_fbank_cos	58.21	66.08	72.11	72.42	21.84	22.45	24.36	23.44
xVec_fbank_lr	61.70	76.76	82.87	86.34	31.05	35.56	37.76	35.48
xVec_fbank_lplda	65.92	82.34	87.15	89.83	**36.11**	**37.44**	**39.05**	**38.43**

Table 2 comparing the accuracy of different back-ends on xVector

The xVector framework is based on [13] aiming to produce embeddings that generalize to unseen speakers. However, in a closed-set LID task, the xVector can classify directly. It is trained on the same language classes as required for deployment. In this section, direct classification is compared with embeddings extracted from the same system.

In table 2, we can find that the performance using embeddings to train the lr or lplda classifier is much better than using the system directly for classification in out-of-domain. Particularly, the direct system appears to suffer from the limited amount of same training data as the in-domain test set channel. It is 14% better on train_25h while xVec_fbank_lplda is only 10% better than xVec_fbank_lplda on tain_106h. Although it is likely that the direct results could be improved with hyper-parameter tuning and calibration in the backend, this emphasize the scalability of standard xVector approach. Once extracted, xVectors can be fed into the same pipeline used for iVectors, taking advantage of existing classifier and backend technology that assists in domain adaptation and calibration.

6.3 Data Augmentation

In this section, we test the importance of augmenting the iVector and xVector DNN training data. The system iVec_mfcc_2f_lr uses 2-fold superimposed augmentation，and iVec_mfcc_5f_lr uses 5-fold combined augmentation. The system xVec_fbank_5f_lr uses 5-fold superimposed augmentation. In either system, the features are fbank which still uses the same augmentation strategy as described in section 5.4.

system	in-domain				Out-of-domain			
	25h	50h	75h	106h	25h	50h	75h	106h
iVec_mfcc_lr	71.43	84.00	88.05	90.62	31.94	37.28	40.15	37.51

xVec_fbank_lr	61.70	76.76	82.87	86.34	31.05	35.56	37.76	35.48
iVec_mfcc_2f_lr	69.89	76.46	87.83	89.25	25.23	31.27	44.94	46.74
iVec_mfcc_5f_lr	**72.52**	**86.54**	**90.09**	**91.83**	**43.31**	**44.61**	**44.86**	**45.36**
xVec_fbank_5f_lr	**62.05**	**76.89**	**83.07**	**89.89**	**33.57**	**36.43**	**37.97**	**43.73**

Table 3 comparing the accuracy of different data augmentation on xVector

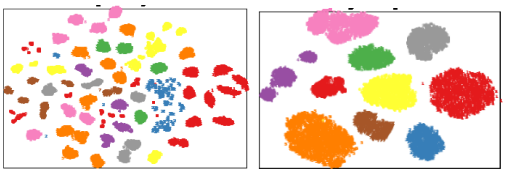
2f: 2-fold superimposed augmentation

5f: 5-fold combined augmentation

In Table 3, we observe that augmentation using 2-fold significantly degrades in in-domain, which may have been corrupted by noise due to raw data. And removing augmentation degrades performance significantly. Due to augmentation increasing the limited amount of training data, the system is more robust against degraded audio. This result parallels training xVectors for speaker recognition in [16].

6.4 BNFs Analysis

The goal is to maximize the distinction between different languages of the xVector system. Obviously, the system only focuses on the inter-class dispersion of language, and ignores the intra-class cohesion of the language. The learned language features have the problem of intra-class divergence as shown in Figure(a). So it attempts to introduce prior knowledge or constraints in the network training process to keep the structure of the basic model as constant as possible, and further enhance the characterization ability of the learned language features. In this paper, the phoneme information is introduced, so that the linguistic features are compensated for the prior knowledge of the phoneme in the learning process to solve the problem of the volatility of the linguistic features caused by the pronunciation content and the speaker. In Figure (a) and (b), the BNFs in our tasks makes each language more convergent and distinguishing.



(a) (b)

Figure 1 The effect of BNFs on the distribution of the extracted features(best viewed in color). The figure shows t-sne visualizations of the xVectors embeddings (a) in case when xVector-fbank (b) in case when xVector-BNF.

system	in-domain				out-of-domain			
	25h	50h	75h	106h	25h	50h	75h	106h
xVec_fbank_lr	61.70	76.76	82.87	86.34	31.05	35.56	37.76	35.48
xVec_fbank_5f_lr	62.05	76.89	83.07	89.89	33.57	36.43	37.97	43.73
xVec_enbnf_lr	**93.72**	**97.66**	**98.31**	**98.41**	**59.13**	**60.78**	**61.02**	**64.22**
~~Fus~~								

Table 4 Comparing Fus with others. ~~Fus represent the combination of xVec_fbank_5f and xVec_enbnf.~~

In Table 4, We observed that only a single English ASR model was used during the experiment. The accuracy of in-domain is nearly 52% better than the xVec_fbank in train_25h. And the accuracy of train_25h is only about 4% lower than train_106h. It seems that the BN feature combined with the xVector system have solved the problem of low-resource. In out-of-domain, the accuracy of xVec_enbnf is 90% better than xVec_fbank in train_25h. It can be seen that the addition of shared language information greatly solves the low-resource and multi-domain problem.

~~Finally, the impact of data augmentation on xVec_enbnf is also explored. Experiments show that adding augmentation is 2% better than removing augmentation.~~

System	en		cn	
	in-domain	out-domain	in-domain	out-domain
fbank	61.7	31.05	61.7	31.05
output-xent.linear	93.72	**59.13**	**96.81**	**64.51**
output.linear	94.04	56.5	96.51	53.87
prefinal-l	**94.35**	55.09	96.53	57.85
tdnn8l	88.96	45.25	89.83	40.6

Table 5 Comparing different layers, under different ASR models

In Table 5, we also compare the effects of different extraction layers of different language models to the LID. It shows that the performance of Chinese ASR model is better than the English ASR model. Because the Chinese ASR model has more training data and the accuracy of phone recognizer is higher. In out-of-domain, BNFs extracted from output-xent.linear is best regardless of the ASR model.

6.5 Multi-language BNFs

It has been well documented that xVector-based LID systems improve the accuracy greatly by using the phoneme information extracted from ASR model. In this section, we show the performance of multi-language BN by comparing systems trained on English BNFs(xVec_enbnf) and Chinese BNFs(xVec_cnbnf). The description of these features is shown in Section 5.2.

system	in-domain				out-of-domain			
	25h	50h	75h	106h	25h	50h	75h	106h
xVec_enbnf_lr	93.72	97.66	98.31	98.41	59.13	60.78	61.02	64.22
xVec_cnbnf_lr	96.81	98.53	98.65	98.91	64.51	64.53	66.40	68.99
Multi-language BN	**97.62**	**98.98**	**98.99**	**99.06**	**65.02**	**65.22**	**68.95**	**70.14**

Table 6 Comparing xVector performance using Multilingual BNFs.

In Table 6, we find that Multi-language BNFs much better than single-language BNFs. In out-of-domain of train_75h, the accuracy of Multi-language BNFs is 13% better than EN-BNFs.

6.6 Domain adaptation

The BNFs maps each language to the phone-related subspace. The distribution of in-domain and out-of-domain data under the BNFs is shown in Figure 2. We can find that the edge distribution of in-domain and out-of-domain data is different. However, the out-of-domain data is equivalent to the convergence of the in-domain data if there are no four languages, 0, 3, 4, and 6.

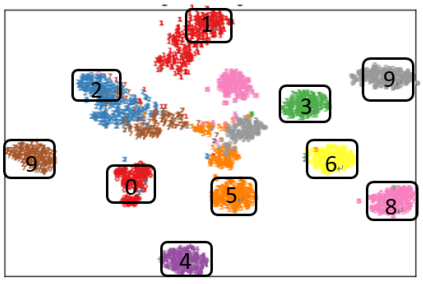


Figure 2 The distribution of xVector-BNFs. The frame represents the in-domain data.

0-Kazak, 1-Tibet, 2-Uyghu, 3-ct-cn, 4-id-id, 5-ja-jp, 6-ko-kr, 7-ru-ru, 8-vi-vn, 9-zh-cn

system	in-domain				out-of-domain			
	25h	50h	75h	106h	25h	50h	75h	106h
xVec_enbnf_lr	93.72	97.66	98.31	98.41	59.13	60.78	61.02	64.22
Train+12min	98.90				79.85			
Adapt	**94.33**	**95.43**	**96.80**	**98.56**	**92.61**	**92.95**	**93.22**	**94.13**

Table 7 Comparison of the performance using adaptation domain,

or adding 12min of annotated out-of-domain data to the training data directly

In Table 7, the adaptation domain method is much better than the others systems in out-of-domain when it is little lower than the others system in in-domain.

7. Conclusion

In this paper, we investigate on how to improve the performance of LID system under low-resource and multi-domain conditions. Two methods is proposed: One is multi-language BN, which is inspired by the advantage and complementarity of universal speech attributes to language-dependent phonemes. The other is the adaptation domain, which allows large-scale training based on large amount of annotated data in the source domain and little amount of annotated data in the target domain. Although the framework can classify directly, we find that this method extracting xVectors from the DNN and using them as features for lr classifier produces much better results in low-resource and multi-domain condition. We explore several variations of the basic xVector framework. We find that bottleneck features improve the performance greatly over acoustic features. Echoing similar results in speaker recognition, our experiments showed that augmenting the xVector DNN training data was a good choice. Finally, We explored the distribution of in-domain and out-of-domain. We find that the relatively distribution of languages in different channels is invariant.

Reference

[1] A. Martin and J. S. Garofolo, “Nist speech processing evaluations: LVCSR, speaker recognition, language recognition,” in Signal Processing Applications for Public Security and Forensics, 2007. SAFE ’ 07. IEEE Workshop on, April 2007, pp. 1 – 7.

[2] Nist language recognition evaluations. [Online]. Available: http://nist.gov/itl/iad/mig/lre.cfm

[3] M.A.Zissman, “Comparison of four approaches to automatic language identification of telephone speech,” IEEE Trans. on Speech and Audio Processing, vol. 4, no. 1, p. 31, 1996. [4] H. Li, B. Ma, and K. A. Lee, “Spoken language recognition: From fundamentals to practice,” Proceedings of the IEEE, vol. 101, no. 5, pp. 1136 – 1159, May 2013.

[5] P. A. Torres-Carrasquillo, E. Singer, M. A. Kohler, R. J. Greene, D.A.Reynolds, andJ.R.DellerJr, “Approaches to language identification
Using Gaussian mixture models and shifted delta cepstral features.” in Proc. Interspeech, September 2002.

[6] M. A. Zissman and E. Singer, “Automatic language identification of telephone speech messages using phoneme recognition and n-gram modeling,” inProc ICASSP-94, vol.1, April1994, pp.305 – 308.

[7] S. M. Siniscalchi, T. Svendsen, and C.-H. Lee, “Toward a detector-based universal phone recognizer,” in Proc. ICASSP, 2008, pp. 4261 – 4264.

[8] C.-H. Lee, “From knowledge-ignorant to knowledge-rich modeling: a new speech research paradigm for next generation automatic speech recognition,” in Proc. Interspeech, 2004, pp. 109 – 112.

[9] AP18-OLR Challenge: Three Tasks and Their Baselines

[10] S. M. Siniscalchi, J. Reed, T. Svendsen, and C.-H. Lee, “Exploiting context-dependency and acoustic resolution of universal speech attribute models in spoken language recognition,” in Proc. Interspeech, 2010, pp. 2718 – 2721.

[11] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motl´ ı ˇ cek, Y. Qian, P. Schwarz, et al., “The Kaldi speech recognition toolkit,” in Proceedings of the Automatic Speech Recognition & Understanding (ASRU) Workshop, 2011.

[12] N. Dehak, P. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, “Front-end factor analysis for speaker verification,” IEEE Trans. on Speech and Audio Processing, vol. 19, pp. 788 – 798, 2011

[13] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, “X-vectors: Robust dnn embeddings for speaker recognition,” in 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2018.

[14] T.Ko, V.Peddinti, D.povey, and S.Khudanpur,”Audio augmentation for speech recognition.,” in Interespeech, 2015, pp. 3586-3589.

[15] D. Snyder, G Chen, and D. Povey, “MUSAN: A Music, Speech, and Noise Corpus,” 2015, arXiv:1510.08484v1.

[16] T.Ko, V.Peddinti, D.Povey, M.Seltzer, and S.Khudanpur, “A study on data augmentation of reverberant speech for robust speech recognition,” in Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on. IEEE, 2017, pp. 5220 – 5224.