The NTU-XJU System for the AP20-OLR Challenge

Jicheng Zhang[1], *Yizhou Peng*[1], *Haobo Zhang*[1], *Haihua Xu*[2], *Hao Huang*[1], *Eng Siong Chng*[2,3]

[1]School of Information Science and Engineering, Xinjiang University, Urumqi, China
[2]Temasek Laboratories, Nanyang Technological University, Singapore
[3]School of Computer Science and Engineering, Nanyang Technological University, Singapore

haihuaxu@ntu.edu.cn, hwanghao@gmail.com

Abstract

In this paper, we present our NTU-XJU system for the oriental language recognition (OLR) challenge, AP20-OLR. The challenge this year contained three tasks: (1) cross-channel LID, (2)dialect identification and (3) noisy LID. We implemented only one network for all Tasks, which is x-vector, with pseudo labeling techniques and multiple data augmentation methods. For task 1, we achieved Cavg values of 0.169 on given dev set defined by official committee of AP20-OLR, and Cavg value of 0.074 on our own defined dev set from experimental data of former challenges for task 2.

Index Terms: AP20-OLR, language identification, x-vector

1. Introduction

The language identification (LID) refers to identify the language categories from utterances, and it is usually realised by methods from speaker verification or speaker identification, such as i-vector [1], x-vector [2] or End-to-End (E2E) neural network classifier based methods [3]. However, there are still difficulties which would decay the performance of LID systems, such as the cross-channel condition, low-resource languages and the noisy environment.

As mentioned in [4], oriental languages often include Austroasiatic languages (e.g.,Vietnamese, Cambodia), Tai-Kadai languages (e.g., Thai, Lao), Hmong-Mien languages (e.g., some dialects in south China), Sino-Tibetan languages (e.g., Chinese Mandarin), Altaic languages (e.g., Korea, Japanese) and Indo-European languages (e.g., Russian).

Besides, dialects can be regarded as different languages in LID tasks as their pronunciations are totally different from each other though they are in one language family.

The oriental language recognition (OLR) challenge is organized annually, aiming at improving the research on multilingual phenomena and advancing the development of language recognition technologies. All data we use in this challenge are from AP16-OLR, AP17-OLR, AP18-OLR, AP19-OLR and AP20-OLR. They include 10 different languages which are Cantonese, Mandarin, Indonesian, Japanese, Russian, Korean, Vietnamese, Kazakh, Tibetan and Uyghur, 3 dialects include Hokkien, Sichuanese and Shanghainese, and also some other nontarget languages such as Malay, Thai, Catalan, Greek and Telugu.

The challenge this year contains three tasks: (1) cross-channel LID, as in the AP19-OLR challenge, (2) dialect identification, where three dialect resources are provided for training, but other three languages are also included in the test set, to compose the open-set dialect identification, and (3) noisy LID, which reveals another real-life demand of speech technology to deal with the low SNR condition.

In the rest of the paper, we will present the data definition for the three tasks respectively, our system description for all tasks, and the experimental results on dev set for tasks 1 and 2.

2. Data definition

Followed by descriptions of [4], we can use data sets as followed to build our language identification models.

- AP16-OL7: The standard database for AP16-OLR, including
- AP16-OL7-train, AP16-OL7-dev and AP16- OL7-test.
- AP17-OL3: A dataset provided by the M2ASR project, involving three new languages. It contains AP17-OL3-train and AP17-OL3-dev.
- AP17-OLR-test: The standard test set for AP17-OLR. It contains AP17-OL7-test and AP17-OL3-test.
- AP18-OLR-test: The standard test set for AP18-OLR. It contains AP18-OL7-test and AP18-OL3-test.
- AP19-OLR-dev: The development set for AP19-OLR. It contains AP19-OLR-dev-task2 and AP19-OLR-dev-task3.
- AP19-OLR-test: The standard test set for AP19-OLR. It contains AP19-OL7-test and AP19-OL3-test.
- AP20-OLR-dialect: The newly provided training set, including three kinds of Chinese dialects.
- THCHS30: The THCHS30 database (plus the accompanied resources) published by CSLT, Tsinghua University [5].

2.1. Settings for development system

For tasks 1 and 2, we subset training, enrollment and evaluation sets from all data mentioned above for building development system.

Task 1, which requires to classify six languages (Mandarin, Japanese, Russian, Vietnamese, Tibetan and Uyghur) where training set and evaluation set are recorded with different devices and environments, is mentioned as cross-channel LID challenge. Task 2 is a dialect identification challenge, which requires to classify three dialects (minnan, shanghai, sichuan) with three nontarget (interfering) languages (Catalan, Greek and Telugu). Table 1 shows the data definition of development systems on tasks 1 and 2.

For task1, we follow the guidance of [4], setting evaluation data as AP19-OLR-test-task2, and we set enrollment data as AP19-OLR-dev-task2 which is different from [4]. And training data is set to all other data available, which includes totally sixteen languages. For task2, we subset AP20-OLR-dialect to

TASKS	TRAINING	ENROLL	EVAL
Cross-channel (Task1)	AP16-OL7 & AP17-OL3 & AP18-OLR-test & AP19-OLR-dev-task3 & AP19-OLR-test-task3 & AP20-OLR-dialect & THCHS30	AP19-OLR-dev-task2	AP19-OLR-test-task2
Dialect (Task2)	AP16-OL7 & AP17-OL3 & AP18-OLR-test & AP19-OLR-dev-task2 & AP19-OLR-test-task2 & THCHS30	AP20-OLR-dialect-enroll	AP20-OLR-dialect-dev & AP19-OLR-test-task3

Table 1: *Settings for development system*

TASKS	TRAINING	ENROLL	EVAL
Cross-channel (Task1)	AP16-OL7 & AP17-OL3 & AP18-OLR-test & AP19-OLR-dev & AP19-OLR-test & AP20-OLR-dialect & THCHS30	AP19-OLR-dev-task2 & AP19-OLR-test-task2	AP20-OLR-test-task1
Dialect (Task2)	AP16-OL7 & AP17-OL3 & AP18-OLR-test & AP19-OLR-dev & AP19-OLR-test & AP20-OLR-dialect & THCHS30	AP20-OLR-dialect	AP20-OLR-test-task2
Noisy (Task3)	AP16-OL7 & AP17-OL3 & AP18-OLR-test & AP19-OLR-dev & AP19-OLR-test & AP20-OLR-dialect & THCHS30	AP16-OL7-NOISY-PART	AP20-OLR-test-task3

Table 2: *Settings for submitted system*

DATASET	DUR(h)	#UTT	#SPK
AP20-OLR-dialect	38	26k	22+15+15
AP20-OLR-dialect-enroll	3	2.1k	11+8+8
AP20-OLR-dialect-dev	2	1.5k	11+7+7

Table 3: *Subsets of AP20-OLR-dialect*

DATASET	DUR(h)	#UTT	#SPK
Noisy (ko-kr & ru-ru)	10.4	7.2k	24+24
Merged Noisy (ja-jp & zh-cn & ct-cn)	14	11k	4+4+4

Table 4: *Data composition in AP16-OL7-NOISY-PART*

AP20-OLR-dialect-enroll and AP20-OLR-dialect-dev by randomly select speakers from each dialect. As shown in table 3, we choose 11, 8 and 8 speakers from Hokkien, Sichuanese and Shanghainese respectively for enroll set, and 11, 7 and 7 speakers for dev set. Speakers chosen for enroll and dev set are not overlapped. So we combine AP20-OLR-dialect-dev and AP19-OLR-test-task3 as which include target languages (Hokkien, Sichuanese and Shanghainese) and interfering languages (Catalan, Greek and Telugu) as dev set of task2 to meet the same condition as test.

2.2. Settings for submitted system

For all tasks, we subset enrollment data from all data available, training set is set to all the data, and test sets for three tasks were given by committee.

As shown in table 2, we combine AP19-OLR-dev-task2 and AP19-OLR-test-task2 as enrollment set for task1, and all data from AP20-OLR-dialect as enrollment set for task2. For task3, we extract some noisy data from AP16-OL7, which include half data of languages of Korean and Russian. We also extract all data from AP16-OL7 of randomly chosen 4 speakers on each of target languages (Cantonese, Japanese, and Mandarin), and merge noisy part of Korean and Russian data into them, where we treat noisy part data as background noise. Finally, as shown in table 4, we combine those merged noisy data with noisy part of Korean and Russian data as enrollment set of task3.

3. Experiment setup

We perform x-vector experiments for all the three tasks. For each experiment, we choose logistic regression as classifier. As we find it consistently yields better results, compared with PLDA [6] method in our case. To train x-vector extractor, we follow the configuration of [2], and the resulting x-vector dimension is 512.

We also perform 2 kinds of data augmentation methods, one is speed perturbation (sp) (0.9x & 1.1x) [7], and the other is spectral augmentation (SpecAugment) [8] with hyper parameters of freq-max-proportion=0.3, time-zeroed-proportion=0.2, and time-mask-max-frames=20.

For further improvement, we also introduce pseudo-labeled technique on tasks 1 and 3, as task 2 contains non-target data which may yield degraded results. We set the proportion of pseudo labels to 0.55 as we achieve the best results on this configuration.

4. Results

We test our x-vector system with sp and SpecAugment on both task 1 and task 2, with experiment setup mentioned above. As shown in table 5, our system achieves EER of 17.97% and Cavg of 0.1693 on dev set of Task1, EER of 13.88 and Cavg of 0.0744 on dev set of Task2.

TASKS	EER%	Cavg
Cross-channel (BL-x-vector)	36.37	0.3583
Cross-channel (Task1)	17.97	0.1693
Dialect (Task2)	13.88	0.0744

Table 5: *Results on development systems*

As we use the same dev data configuration with official

baseline system on cross-channel challenge, we find our system achieves more than 50% of relative improvement (18.5% of absolute improvement) compared with baseline x-vector system on task1.

5. Conclusion

In this paper, we briefly introduced our x-vector systems with pseudo-labeled techniques and some data augmentation methods on AP20-OLR challenge, and we redefined all data available to some degree for our own experiment and research. We achieved more than 50% relative improvement (18.5% of absolute improvement) on development system of task 1 compared with official baseline x-vector system.

6. References

[1] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE TASLP*, vol. 19, no. 4, pp. 788–798, 2011.

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

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

[4] Q. H. L. L. Z. T. D. W. L. S. Zheng Li, Miao Zhao and C. Yang, "Ap20-olr challenge: Three tasks and their baselines," *arXiv preprint arXiv:2006.03473*, 2020.

[5] D. Wang and X. Zhang, "Thchs-30: A free chinese speech corpus," *arXiv preprint arXiv:1512.01882*, 2015.

[6] Y. Wang, H. Bai, M. Stanton, W.-Y. Chen, and E. Y. Chang, "Plda: Parallel latent dirichlet allocation for large-scale applications," in *International Conference on Algorithmic Applications in Management*. Springer, 2009, pp. 301–314.

[7] T. Ko, V. Peddinti, D. Povey, and S. Khudanpur, "Audio augmentation for speech recognition," in *Proc. of INTERSPEECH 2015*, 2015.

[8] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, "SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition," in *Proc. of INTERSPEECH 2019*, pp. 2613–2617.