CROSS-VALIDATION OF MULTIPLE LANGUAGE RECOGNITION SYSTEMS USING PSEUDO KEYS

Hanwu Sun, Bin Ma and Haizhou Li

Institute for Infocomm Research (I²R), A*STAR, 1 Fusionopolis Way, Singapore 138632

{hwsun, mabin, hli}@i2r.a-star.edu.sg

ABSTRACT

In this paper, we present a pseudo-key analysis approach for cross-validation of language recognition systems before the ground truth (true key) becomes available. A state-of-the-art language recognition system typically employs multiple language recognition classifiers which are fused to form a mixture of experts. The individual classifiers are also called subsystems. To avoid the fused system from being brought down by some outlier classifiers, pseudo keys are designed to cross-examine the integrity of individual classifier candidates. The language recognition experiments are conducted on the NIST 2007 Language Recognition Evaluation (LRE) corpus using the subsystems in the primary submission from the Institute for Infocomm Research (IIR).

Index term: Language recognition, NIST language recognition evaluation, language, design.

1. INTRODUCTION

Spoken language recognition is the process of determining the identity of the language in a spoken utterance. It is one of the enabling technologies in speech applications such as multilingual speech recognition, speech translation, and spoken document retrieval. In the past decade, the National Institute of Standards and Technology (NIST) has conducted a series of technology evaluation [1]. The Language Recognition Evaluations (LRE) focus on language and dialect detection of telephony conversational speech.

A good language recognition system exploits discriminative cues of spoken languages from multiple resources. Acoustic and phonotactic features are believed to be complementary in language characterization [2, 3], which are used in different individual classifiers. To fuse multiple individual classifiers, we need to decide how to weight them according to their performance. This can be achieved by estimating the weights from a development dataset which has the ground truth (true key) and is close enough to the actual test dataset.

In the NIST LREs, sufficient training and development data with the true key are available to train the individual language classifiers as well as the fusion weights. This only guarantees that all the classifier candidates and their fusion work well on the development dataset, but not necessarily on the evaluation dataset as the true keys are simply not available. If one of the classifiers malfunctions for whatever reasons, the fused system may be brought down unexpectedly. Are we able to get an idea of how the individual classifiers work without having the ground truth?

We assume that if two individual classifiers are about the same competent, then their language recognition decisions should be considerably consistent. Given a collection of classifiers, we propose to derive a set of pseudo-key from one classifier and to use the pseudo-key to cross-validate the other classifiers.

We discuss the IIR system in NIST 2007 LRE in Section 2. We introduce the pseudo-key analysis method in Section 3. Section 4 reports the experiments. Finally we conclude in Section 5.

2. LANGAUGE RECOGNITION SYSTEM

2.1. Individual Language Classifiers

The IIR primary system in the NIST 2007 LRE [4] is a fusion of eleven language classifiers, of which six are based on phonotactic features, while five others are based on acoustic features.

A phonotactic classifier adopts a set of parallel phone recognizers (PPR) in the front-end that converts a spoken utterance into sequences of phones. We have PPR-LM [5], PPR-VSM [6], TOPT-VSM [7], PAD-PPR-VSM [8] using seven PPRs developed in IIR, and BUT-PPR-LM and BUT-PPR-VSM using PPRs developed by the Brno University of Technology (BUT)[1].

The five acoustic classifiers include ML-GMM using maximum likelihood (ML) training for GMM modeling, MMI-GMM adopting maximum mutual information (MMI) training [9], MFCC-GLDS applying the generalized linear discriminate sequence kernel (GLDS) [10] for the SVM modeling based on MFCC features, LPCC-GLDS applying GLDS SVM based on LPCC features, and PSK-SVM [11] adopting probabilistic sequence kernel (PSK) for SVM modeling.

[1] http://www.fit.vutbr.cz/research/groups/speech/index_e.php?id=phnrec

2.2. System Fusion

The final system is obtained by means of linear fusion of the scores from the eleven classifiers:

$$s_i = \sum_{f=1}^{N} w_f s(f,i) + b \,, \tag{1}$$

where $N=11$ is the total number of classifiers and $s(f,i)$ is the score of the i-th language recognition trial from the f-th classifier. The fusion parameters consist of the classifier specific weights w_f and the global bias b. We use a minimum equal error rate (EER) objective to tune the fusion parameters:

$$(w_f, b)_{\min EER} = \arg\min_{w_f, b} \{ \| P_{miss} + P_{FA} \| \subset P_{miss} = P_{FA} \} \,, \tag{2}$$

where the miss-detect (*miss*) and false-alarm (*FA*) probabilities are given in Eq.(3).

$$\begin{aligned} P_{\text{miss}} &= \frac{|\{i : i \in \text{True}, s_i < b\}|}{|\{i : i \in \text{True}\}|} \\ P_{\text{FA}} &= \frac{|\{i : i \in \text{False}, s_i \geq b\}|}{|\{i : i \in \text{False}\}|} \end{aligned} \tag{3}$$

and $|\{\ldots\}|$ denotes the size of the dataset.

3. PSEUDO KEY

3.1. Pseudo Key by Peer Review

Each of the classifiers in the fused system is built in multiple steps, such as front-end feature extraction, channel and session variability compensation and speaker modeling. It is possible that some steps may go wrong for some reasons. Before the true keys of the evaluation trials are available, it is desirable to check the integrity of each individual classifier before the system fusion. We propose to derive pseudo keys from the classifier candidates themselves, and use them as if they were the true keys. With the pseudo-key analysis, one is able to cross-validate the scores between a pair of classifiers. This is equivalent to peer review, thus is also called Peer Review (PR) pseudo-key.

A classifier is typically formulated as a hypothesis test. For each target language l, we build a language detector which consists of two language models $\{l^+, l^-\}$. l^+ is trained on the data of target language, while l^- is trained on the data of the competing languages. We define the confidence of a test sample O belonging to language l^+ as the posterior odds in a hypothesis test under the Bayesian interpretation. We have H_0, which hypothesizes that O is language l^-, and H_1, which hypothesizes otherwise. The posterior odd is approximated by the likelihood ratio λ that is used for the final language recognition decision:

$$\lambda = \log\left(\frac{p(O|l^+)}{p(O|l^-)} \right) . \tag{4}$$

Suppose that there are M trials for the L target languages. In the closed-test, the genuine/imposter ratio is given by 1:(L-1), From the pool of scores of M trials from each classifier, f, we consider the M/L trials with highest scores as the genuine trials and the remaining trials as the impostor trials, thus having the Peer Review (PR) pseudo-key as,

$$\tilde{k}(f,i) = \begin{cases} \text{True}, & \text{if } s(f,i) \geq \tilde{\lambda}_f \\ \text{False}, & \text{if } s(f,i) < \tilde{\lambda}_f \end{cases} , \tag{5}$$

where $\tilde{k}(f,i)$ denotes the pseudo-key for the i-th trial of the f-th classifier and the threshold $\tilde{\lambda}_f$ is set such that there are M/L trials whose scores are above it. In the above equation, $s(f,i)$ represent the score of the i-th trial from the f-th classifier. Using the pseudo keys from all other classifiers, we can compute the pseudo EER for the f-th classifier as

$$EER_{Pseudo}(f) = \frac{1}{N-1} \sum_{g=1, g\neq f}^{N} EER\left\{ s(f,i) \,|\, \tilde{k}(g,i) \right\} , \tag{6}$$

where $EER\{\cdot\}$ is the operator to obtain EER, and N is the total number of classifiers.

As the pseudo keys and true keys are inevitably different, there will be some discrepancies between the actual EER and pseudo EER. Assuming the development dataset is similar to the evaluation dataset, we can know the exact difference between the actual EER and the pseudo EER on the development dataset, the discrepancy presented in the evaluation dataset can therefore be estimated. If a classifier works well on both the development and evaluation dataset, then we expect to observe the same EER discrepancy on both datasets. The estimated pseudo EER on the evaluation data can be adjusted as:

$$EER_{Pseudo}(f)|_{Eval} = \frac{1}{N-1} \sum_{g=1, g\neq f}^{N} EER\left\{ s(f,i) \,|\, \tilde{k}(g,i) \right\} + \Delta EER(f)|_{Dev} , \tag{7}$$

and

$$\Delta EER(f)|_{Dev} = EER_{Actual}(f)|_{Dev} - EER_{Pseudo}(f)|_{Dev} \quad .$$

where the subscript '*Dev*' and '*Eval*' stand for the development and evaluation dataset, respectively.

3.2. Pseudo Key by Jury Panel

After we derive the fusion weights on the development dataset, we can apply them in the development and evaluation datasets to obtain their fused systems. These final fused systems or scores can act as a mixture of experts – Jury Panel to generate a set of pseudo-key – Jury Panel (JP) pseudo-key in a similar way to that for PR pseudo-key. We define the pseudo EER for the f-th classifier based on the JP pseudo-key approach as,

$$EER_{Pseudo}(f)|^{F} = EER\left\{ s(f,i) \,|\, \tilde{F}(i) \right\} \tag{8}$$

where $\widetilde{F}(i)$ denotes the JP pseudo-key for the i-th trial as defined in Eq.(5).

Considering the discrepancies between the actual EER and pseudo EER, we can similarly adjust the pseudo EER rates for the evaluation dataset as:

$$EER_{Pseudo}(f)|_{Eval}^{F} = EER\left\{s(f,i)\,|\,\widetilde{F}(i)\right\} + \Delta EER(f)|_{Dev}^{F}, \quad (9)$$

and

$$\Delta EER(f)|_{Dev}^{F} = EER_{Actual}(f)|_{Dev} - EER_{Pseudo}(f)|_{Dev}^{F}.$$

where $EER_{Pseudo}(f)|_{Dev}^{F}$ is the pseudo EER of f-th classifier on the development dataset by using JP pseudo-key.

4. EXPERIMENTS

We evaluate the pseudo-key analysis on the NIST 2007 LRE General LR 30-second closed-test tasks. The General Language Recognition (LR) task includes 2,510 trials in 14 target languages. We present the pseudo EER using the proposed PR and JP pseudo-key approaches, and study how one outlier system affects the results of the final fused system.

4.1. Results with PR Pseudo Key

We apply the pseudo-key approach to analyze the performance of eleven classifiers on the NIST 2007 LRE development data set as well as the evaluation dataset. The pseudo EER is computed using Eq.(6) and Eq.(7). Figure 1 compares the pseudo EER and actual EER for all the eleven classifiers in the closed-test task. It is shown that the pseudo EER and actual EER on both the development and evaluation data sets are generally consistent, and the pseudo EER therefore can provide a good indication of the performance of the classifiers.

To confirm the reliability of pseudo-key analysis, we carry out the T-test [12] for the confidence level test between the actual EER and pseudo EER on the NIST 2007 LRE closed-test task. For the development dataset, its significance/probability (at 5% significance level two-tailed test) between pseudo EER and actual EER is 94.29% and its 95% confidence interval on the estimated EER is [-1.0503, 1.0973]. Similarly, we can get the T-test results for the evaluation data set. The significance is 95.55% and its 95% confidence interval on the estimated EER is [-1.6031, 1.5832]. Obviously, we achieve similar significance levels in the T-test on both the development and evaluation dataset for the closed-test task. In other words, if a classifier behaves on the evaluation dataset very differently from the development dataset in the PR pseudo-key analysis, then we have a good reason to think that this classifier is in disorder.

4.2. Results with JP Pseudo Key

We continue the experiments on the same task using JP pseudo-key as in Eq.(8) and Eq.(9). The pseudo EER and actual EER are shown in Figure 2, which compares the pseudo EER and actual EER for all the eleven classifiers in the closed-test task. We observed that the pseudo EER and actual EER on both the development and evaluation datasets using JP pseudo-key are generally more consistent than the results shown in Figure 1. It is because the JP pseudo-key is closer to the true key than the PR pseudo-key. Of course, this is based on the assumption that the fused system works better than any individual systems.

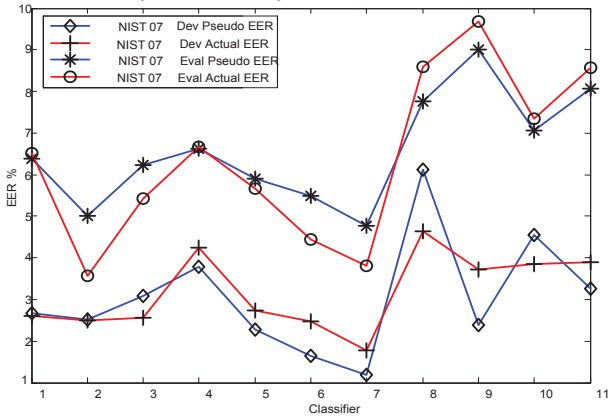

Figure 1: Pseudo and Actual EERs Evaluated on the Development and Evaluation Sets of the NIST 2007 LRE (30s General LR close-test condition) Using PR Pseudo-key

We also conduct that the T-Test on the development dataset based on the JP pseudo-key, the significance/probability (at 5% significance level two-tailed test) between pseudo EER and actual EER is 95.48% and its 95% confidence interval on the estimated EER is [-0.8326, 0.8273]. For the evaluation dataset, the significance is 96.52% and its 95% confidence interval on the estimated EER is [-1.2409, 1.2388]. As expected, the 95% confidence intervals on the estimated EER for both development data and evaluation data using JP pseudo-key are smaller than those using PR pseudo-key.

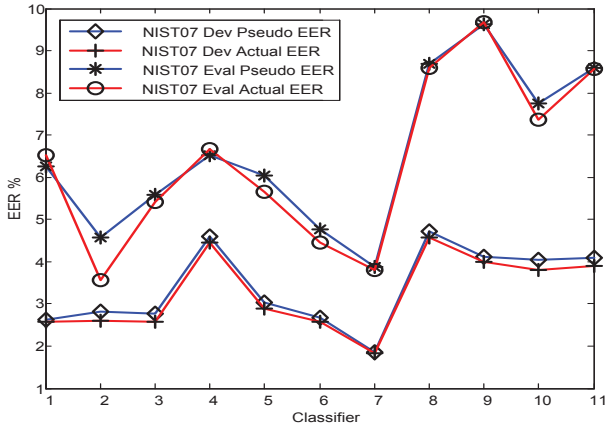

Figure 2: Pseudo and Actual EERs Evaluated on the Development and Evaluation Datasets of the NIST 2007 LRE (30s General LR open-test condition) Using JP Pseudo-key

4.3. Simulation with Disorder Classifier

In this section, we study how a disorder classifier can be spotted through the pseudo-key analysis and how it affects the fused system. The minimum EER objective function in Eq.(2) is used to estimate the fusion weights. In order to simulate a disorder classifier, we randomly choose a classifier and manually reduce the score values of genuine trials while keeping the score values of imposter trials unchanged:

$$s(f,i)_{downgrade} = \begin{cases} s(f,i) - C & i^{th} \text{ trial } is \text{ genuine} \\ s(f,i) & \text{otherwise} \end{cases}, \quad (10)$$

where C is a positive constant. One can expect larger C leads to poorer EER performance. By changing C, we can obtain various simulated EERs for a classifier.

We pick the 7th classifier as the outlier classifier while keeping the remaining 10 classifiers unchanged. We increase C in Eq.(10) gradually and train the fusion weights following Eq.(2). We report the results of the 7th classifier on the NIST 2007 LRE evaluation dataset in Figure 3.

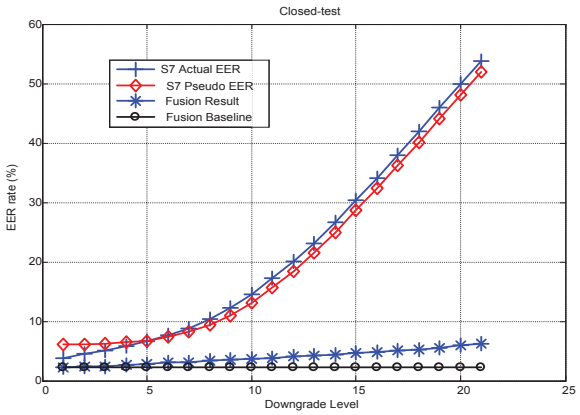

Figure 3: The EER Performance of the 7th Classifier (S7) and Fused System as a Function of Various Downgrading Level of S7 on NIST 2007 LRE Closed-test (30s General-LR task).

Table 1: The EER Simulation for the Fused System Affected by the 7th Classifier Benchmarked Against the True Keys

	EER %				
	Baseline	Simulated Downgrading Scores			
Closed test Classifier 7	3.81	5.75	8.73	23.11	45.95
Closed test Fusion	2.19	2.59	3.10	4.22	5.51

From Figure 3, we can see that the pseudo EER is generally consistent with the actual EER using PR pseudo-key. This also shows that the proposed pseudo-key analysis can work well for wide range EER conditions. It is not surprising to find that the fused system's performance degrades as the EER of the 7th classifier increases. One outlier classifier can greatly affect the final fused system. This can be spotted through a pseudo-key analysis by detecting abnormal pseudo EER without the need of ground truth (actual EER). Table 1 shows the simulation for the fused system under various downgrading levels of the 7th classifier using the true keys on the NIST 2007 LRE closed-test 30s test sets.

5. CONCLUSION

In this paper, we studied a novel approach to detect disorder of individual language classifiers for effective fusion. From the experimental results on the closed-test tasks of the NIST 2007 LRE 30-second general language recognition, the EERs predicted with the pseudo-key analysis are reasonably consistent with the actual EERs, especially for the JP pseudo-key. The reliability of pseudo-key analysis has also been confirmed by T-test. A study using simulated disorder classifier shows that an outlier classifier can greatly affect the final fused system, which can be avoided using pseudo-key analysis.

6. REFERENCE

[1] http://www.nist.gov/speech/tests/lang/2007/.

[2] E. Singer, P. A. Torres-Carrasquillo, T. P. Gleason, W. M. Campbell and D. A. Reynolds, "Acoustic, Phonetic and Discriminative Approaches to Automatic Language Recognition," in *Proc. Eurospeech,* 2003.

[3] R. Tong, B. Ma, D. Zhu, H. Li and E. S. Chng, "Integrating acoustic, prosodic and phonotactic features for spoken language identification," in *Proc. ICASSP*, 2006.

[4] The 2007 NIST Language Recognition Evaluation plan, http://www.nist.gov/speech/tests/lang/2007/LRE07EvalPlan-v8b.pdf.

[5] H. Li, B. Ma, and C.-H. Lee, "A Vector Space Modeling Approach to Spoken Language Identification," *IEEE Transaction on Audio, Speech and Language Processing*, Vol. 15, No. 1, 2007.

[6] B. Ma, H. Li, and R. Tong, "Spoken Language Recognition Using Ensemble Classifiers," *IEEE Transactions on Audio, Speech and Language Processing*, Vol. 15, issue 7, pp. 2053-2062, 2007.

[7] R. Tong, B. Ma, H. Li and E. S. Chng, "Target-oriented Phone Tokenizers for Spoken Language Recognition," in *Proc. ICASSP,* 2008.

[8] K. C. Sim and H. Li, "Fusion of Contrastive Acoustic Models for Parallel Phonotactic Spoken Language Identification," in *Proc. Interspeech* 2007.

[9] Burget L., Matejka P. and Cernocky J., "Discriminative training techniques for acoustic language identification," *ICASSP*, 2006.

[10] W. M. Campbell, J. P. Campbell, D. A. Reynolds, E. Singer and P. A. Torres-Carrasquillo "Support vector machines for speaker and language recognition," *Computer Speech and Language*, vol. 20, pp. 210-229, 2006.

[11] K. A. Lee, C. You, and H. Li, "Spoken language recognition using support vector machines with generative from-end," in *Proc. ICASSP*, 2008,

[12] R. E. Walpole and R. H. Myers. *Probability and Statistics for Engineers and Scientists*. Macmillan, Inc., New York, 4th edition, 1989.