INTERACTIVE LEARNING OF TEACHER-STUDENT MODEL FOR SHORT UTTERANCE SPOKEN LANGUAGE IDENTIFICATION

Peng Shen, Xugang Lu, Sheng Li, Hisashi Kawai

Institute of Information and Communications Technology, Japan
peng.shen@nict.go.jp

ABSTRACT

Short utterance-based spoken language identification (LID) is a challenging task due to the large variation of its feature representation. Improving feature representation of short utterances using a teacher-student method has been shown its effectiveness for LID tasks. However, conventional teacher-student methods use fixed pre-trained teacher models, that makes it difficult to optimize student models. In this paper, rather than using a fixed pre-trained teacher model, we investigate an interactive teacher-student learning by adjusting the teacher model with reference to the performance of the student model when the student model is stuck in a local minimum. Experiments on a 10-language LID task were carried out to test the algorithm. Our results showed its effectiveness of the proposed algorithm on short utterance LID tasks.

Index Terms— Interactive teacher-student learning, teacher model optimization, knowledge distillation, short utterance feature representation, spoken language identification

1. INTRODUCTION

Spoken language identification (LID) techniques are important for multilingual applications, such as multilingual automatic speech recognition and translation systems [1, 2]. LID techniques are typically used as a pre-processing stage of multilingual speech applications. For real-time systems, improving the performance of LID on short utterances is one of the important tasks to reduce the real-time factor of the whole system.

One of the state-of-the-art LID approaches is the i-vector-based method. I-vector approaches have been demonstrated their effectiveness and obtained state-of-the-art performance in many LID tasks, especially on relatively longer utterance tasks [3, 4, 5, 6, 7, 8]. In these approaches, compact utterance-level i-vectors were extracted for language feature representation, then classifiers were used for classification. However, the performance of the i-vector-based approaches often degrade dramatically on short utterance LID tasks, one of the main reasons is that the i-vector representation for short utterances has a large distribution variation.

Recently, end-to-end approaches with deep neural networks (DNN), recurrent neural networks (RNN), convolutional neural networks (CNN) and attention-based neural networks have been investigated on LID tasks [9, 10, 11, 12, 13]. Compared with i-vector-based approaches, the end-to-end approaches do not include many hand-crafted algorithmic components which makes it easy to be optimized. For short utterance LID tasks, the end-to-end approaches demonstrated impressive performance [9, 11, 12]. For example, Lopez-Moreno et al. proposed to use frame level-based DNN method for LID tasks and outperformed the conventional i-vector system on short duration utterance tasks, i.e. 3 seconds [9]. A bidirectional long short term memory network (biLSTM) by modelling temporal dependencies features using the past and future frames was proposed for short durations (3 seconds) [12]. Lozano-diez et al. used deep convolutional neural networks (DCNN) for short test durations (segments up to 3 seconds of speech) [11].

Similar to previous work [11], our previous experiments also showed the effectiveness of the DCNN-based end-to-end approach on short utterance LID tasks. However, the performance of the DCNN model decreases rapidly as the input sentence becomes shorter, even the model is completely trained on short utterances [14]. The challenging for short utterance-based LID is the large variations of feature representations. To solve this problem, a feature representation knowledge distillation (FRKD) method was proposed to improve the feature representation of short utterances[14]. The FRKD method used a robust feature representation (obtained with a longer utterance-based teacher model) to normalize the feature representation of a short utterance-based student model. Such normalization can make the student model to mimic the feature extraction behavior of the teacher model.

The FRKD method is motivated by the knowledge distillation method [15]. Knowledge distillation method has been already successfully applied on many tasks, such as speech recognition, image classification [15, 16]. Conventional knowledge distillation methods use fixed pre-trained models as teacher, the performance of the student model depends on how well the student learns knowledge from the teacher. Different from conventional knowledge distillation method with same inputs for teacher and student models, in

FRKD framework, the inputs of the student model are short utterances while the inputs of the teacher model are the corresponding longer utterances. Such difference makes it difficult to optimize the student model with a fixed pre-trained teacher model, and the student model is easy to be stuck in a local minimum with a bad performance. In this work, rather than using a fixed pre-trained teacher model, we investigate an interactive teacher-student learning method to improve the teacher-student learning by adjusting the teacher model with reference to the performance of the student model. To the best of our knowledge, the proposed approach has not yet been studied by other researchers on LID tasks. We evaluated the proposed method on a 10-language dataset. Our results showed its effectiveness of the proposed algorithm on short utterance LID tasks.

2. FEATURE REPRESENTATION KNOWLEDGE DISTILLATION FRAMEWORK

In conventional knowledge distillation method, a high performance teacher model is important, therefore, one big model or ensemble multiple models are often used as the teacher model [15, 17]. For LID tasks, compared with short utterances, the performance of longer utterances is better. The FRKD method was proposed by using the knowledge of a long-utterance-based teacher model for short-utterance-based student model training. In FRKD framework, the teacher's feature representation knowledge is used to regularize the student network, that can help the student network capturing robust discriminative feature for short utterances.

Mathematically, given a short utterance $\mathbf{x}_S$, and its corresponding long utterance $\mathbf{x}_T$. $\mathbf{y}$ is the target label. Let $\mathbf{\Theta}_S$ be parameter sets of hidden layers' feature representation of a student network, where $\mathbf{\Theta}_S = \{\mathbf{W}_S, \mathbf{b}_S\}$. Similarly, let $\mathbf{\Theta}_T$ be hidden layers' parameter sets of a teacher network. Then, the student model can be optimized by minimizing the following loss function:

$$\hat{L}_{\text{FRKD}} = (1-\lambda)L_S(\mathbf{x}_S, \mathbf{y}) + \lambda L_{\text{kt}}(\mathbf{x}_T, \mathbf{x}_S, \mathbf{\Theta}_T, \mathbf{\Theta}_S) \quad (1)$$

where $L_S(\mathbf{x}_S, \mathbf{y})$ is the loss function of the student model, and L_{kt} is the regularization term. Then, for $\mathbf{x}_S$, the cross entropy-based loss can be described as:

$$L_S(\mathbf{x}_S, \mathbf{y}) = -\sum_i y_i \log p_i(\mathbf{x}_S), \quad (2)$$

and, the regularization term can be defined as:

$$L_{\text{kt}}(\mathbf{x}_T, \mathbf{x}_S, \mathbf{\Theta}_T, \mathbf{\Theta}_S) = \|u_T(\mathbf{x}_T; \mathbf{\Theta}_T) - u_S(\mathbf{x}_S; \mathbf{\Theta}_S)\|_1, \quad (3)$$

where $\| \bullet \|_1$ is the L1-norm to evaluate the representation distance between the teacher and student models, and u_T and u_S are the teacher and student deep nested functions up to their respective selected layers. And λ is a weight coefficient. In conventional teacher-student model, the parameters Θ_T of a teacher model is fixed after it was trained.

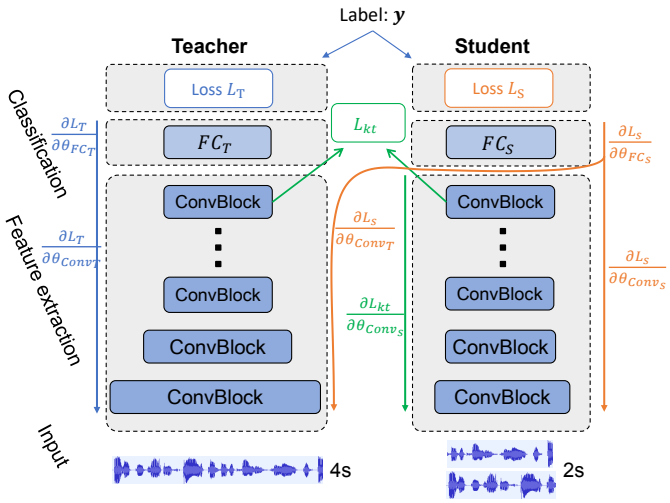

Fig. 1. The proposed interactive teacher-student learning.

3. INTERACTIVE TEACHER-STUDENT LEARNING

Conventional knowledge distillation methods focus on training small/compact models for easy deployment by using fixed pre-trained high-performance models. Because of the input samples for teacher and student models are same, it is reasonable for the student model to mimic the behavior of the teacher model well. Considering the situation of the inputs of teacher and student are different, e.g., the FRKD framework, it will be a challenging task for the student model to learn well because of the difference. As a consequence, the optimization of the student model is easy to be stuck in a local minimum with a bad performance. In this work, we investigate an interactive learning method to improve the conventional teacher-student learning under the FRKD framework. Our idea is based on this consideration: when a student model is stuck in a local minimum, it is possible to drive the optimization to escape from the local minimum with a parameter adjustment on its teacher model. Fig. 1 illustrates the proposed interactive learning framework. The proposed method can be considered as a unified learning framework, that optimizes both the teacher and student models by minimizing the following loss function,

$$\begin{aligned}\hat{L}_{\text{ITSL}} = (1-\alpha-\beta)L_T(\mathbf{x}_T, \mathbf{y}) + \alpha L_S(\mathbf{x}_S, \mathbf{y}) \\ +\beta L_{\text{kt}}(\mathbf{x}_T, \mathbf{x}_S, \mathbf{\Theta}_T, \mathbf{\Theta}_S),\end{aligned} \quad (4)$$

where $L_T(\mathbf{x}_T, \mathbf{y})$ and $L_S(\mathbf{x}_S, \mathbf{y})$ are losses of the teacher and student models, respectively, and L_{kt} is the hidden layer-based feature representation regularization term.

We suppose that the effective learning between teacher and student models is: students learn knowledge from teachers, and teachers should adjust their model parameters to maximize the efficiency of the learning process of their students. Under this assumption, the regularization term, i.e., L_{kt} is

only used to regularize the feature representation of student models. And teacher models are optimized by considering the losses of their student model. Then, for student model optimization, Eq. 4 can be redefined as,

$$\hat{L}_{\mathrm{S}} = (1-\lambda)L_S(\mathbf{x}_S, \mathbf{y}) + \lambda L_{\mathrm{kt}}(\mathbf{x}_T, \mathbf{x}_S, \mathbf{\Theta}_T, \mathbf{\Theta}_S), \quad (5)$$

that is same to Eq. 1. The teacher model is optimized with the following loss function,

$$\begin{aligned}\hat{L}_{\mathrm{T}} = (1-\gamma-\xi)L_T(\mathbf{x}_T, \mathbf{y}) + \gamma L_S(\mathbf{x}_S, \mathbf{y}) \\ +\xi\|u_T(\mathbf{x}_T; \mathbf{\Theta}_T) - u_T^0(\mathbf{x}_T; \mathbf{\Theta}_T^0)\|_1,\end{aligned} \quad (6)$$

where $L_T(\mathbf{x}_T, \mathbf{y})$ can be described as,

$$L_T(\mathbf{x}_T, \mathbf{y}) = -\sum_i y_i \log p_i(\mathbf{x}_T), \quad (7)$$

and $\|u_T(\mathbf{x}_T; \mathbf{\Theta}_T) - u_T^0(\mathbf{x}_T; \mathbf{\Theta}_T^0)\|_1$ is a constraint that keeps the model is not optimized to be too far away to its initial model, where $u_T(\mathbf{x}_T; \mathbf{\Theta}_T)$ and $u_T^0(\mathbf{x}_T; \mathbf{\Theta}_T^0)$ are the representations of updated and initial teacher models, respectively. The parameter changes of the related hidden layers of teacher model will affect the optimization of the student model, because of L_{kt} in Eq. 5. Therefore, $L_S(\mathbf{x}_S, \mathbf{y})$ and $\mathbf{\Theta}_T$ are correlated. In practice, we use a minibatch-based iteration optimization algorithm, i.e., Algorithm 1, for teacher and student models optimization.

Algorithm 1 Interactive teacher-student learning.

Teacher model training:
Given samples $\mathbf{x}_T$ and labels $\mathbf{y}$.
for number of training iterations **do**
    Sample minibatch sample sets from training dataset.
    Pre-train teacher model with $L_T(\mathbf{x}_T, \mathbf{y})$.
end for
Student model training:
Given samples $\mathbf{x}_S$, $\mathbf{x}_T$ and labels $\mathbf{y}$.
for number of training iterations **do**
    Sample minibatch sample sets from training dataset.
    Train student model with $\hat{L}_{\mathrm{S}}$.
end for
Interactive teacher-student training:
Given samples $\mathbf{x}_S$, $\mathbf{x}_T$ and labels $\mathbf{y}$.
for number of (training_iterations $\times$ num_minibatch) **do**
    Sample one minibatch of sample sets.
    Tune the teacher model with $\hat{L}_{\mathrm{T}}$.
    Tune the student model with $\hat{L}_{\mathrm{S}}$.
end for

4. EXPERIMENTS

Experiments were conducted to evaluate the effectiveness of the proposed method. We used a 10-language dataset of NICT to evaluate the proposed method. The data were spoken by native speakers. We split them into training (Train), validation (Valid), and test (Test) sets. There were 100.76 hours of training data, and 24.95 hours of test data. The average duration was 7.6 seconds. The number of utterances for the training data was 45000, and for each language was 4500. For the validation and test data, it was 300 and 1200 utterances for each language. Detailed information can be referred to [14]. The utterance identification error rate (UER) was used as the evaluation criterion.

4.1. Implementation of baseline systems

I-vector-based method with multiclass logistic regression classifier was examined for comparison. The i-vectors were 600-dimensional vectors that extracted with 12-dimensional MFCCs and log power feature applied shifted delta cepstral. The script of Kaldi toolkit [18] was used for the i-vector system preparation.

For end-to-end methods, we built systems with RNN and DCNN for short utterance LID tasks, i.e., 2.0s, 1.5s and 1.0s. The DCNN architecture used for four-second inputs (teacher model) is illustrated in Table 1. The DCNN model included seven convolutional layer blocks and two fully-connected layer blocks. Each convolutional layer block included one convolution layer, one max-pooling layer and one batch normalization layer. The fully-connected layer block included one fully-connected layer and one batch normalization layer. For inputs of different lengths, the stride of max-pooling was changed to make the output of the last convolution layer had same dimension. For comparison, we built systems with RNN and bidirectional RNN and compared different configurations (one or more hidden layers with 256 neurons) and dropout with 0.0, 0.3 and 0.5. The mini-batch size was set to 32, RMSProp optimizer with learning rate 0.001 for model optimization. The maximum learning epoch was set to 100, and the optimal model was selected using the validation data set.

To extract the target length utterances, power energy-based VAD was used to detect the speech, then certain length utterances were cut with a shift that equaled to the duration length. Then, 60-dimensional mel-filterbank features were extracted for all the utterances. Finally, mean and variance normalization was applied. For the testing dataset, only the beginning of the speech was used based on the VAD result.

4.2. Implementation of FRKD and the proposed method

In FRKD framework, we used a four-second-based DCNN model as the teacher model. The output of the flatten hidden layer was used for feature representation regularization, i.e., L_{kt}. The student models were optimized with Eq. 1. For λ, we evaluated it with values of 0.1, 0.3, 0.5 and 0.7.

The proposed method was implemented based on the FRKD framework. The interactive teacher-student training were done based on the models trained with FRKD. There

Table 1. The DCNN network used for the teacher model; same padding was used for conv and max-pooling layers.

Teacher Network(4.0s)
Input: $\mathbf{x} \in \Re^{400\times 60}$
conv (7×7, 16, relu), max-pooling(3×3, stride 2 × 2), BN conv (5×5, 32, relu), max-pooling(3×3, stride 2 × 2), BN conv (3×3, 64, relu), max-pooling(3×3, stride 2 × 2), BN conv (3×3, 64, relu), max-pooling(3×3, stride 2 × 2), BN conv (3×3, 128, relu), max-pooling(3×3, stride 2 × 2), BN conv (3×3, 128, relu), max-pooling(3×3, stride 2 × 2), BN conv (3×3, 256, relu), max-pooling(3×3, stride 2 × 2), BN Flatten()
FC(512, relu), BN FC(512, relu), BN
Output: softmax(10)

were two optimization steps, i.e., minimizing Eq. 5 and Eq. 6. We set λ to 0.3 for Eq. 5 by referring to the investigation of FRKD. For Eq. 6, we fixed ξ to 0.1 and evaluated γ with 0.1, 0.2 and 0.3. Because of the loss of a pre-trained student model on training data will become very small, we used the loss of the validation data for Eq. 6. For all the durations, i.e., 1.0s, 1.5s, and 2.0s, we used the same teacher model. The basic configuration of student models was same besides the stride of max-pooling setting. Other settings, e.g., optimizer, mini-batch, were same to that of the DCNN baseline systems.

4.3. Results and discussions

Table 2 shows the results of i-vector system, RNN, biRNN and DCNN models with two-second utterances. For RNN models, the best results were obtained with two GRUs without dropout, and two biGRUs with dropout set to 0.3. Dropout setting was also investigated on DCNN models, however, we could not obtain further improvement. Compared with other systems, the DCNN model performed the best on this dataset.

For FRKD method, the best result was obtained with λ set to 0.3. Based on the best setting of the FRKD method, we investigated to prevent local minimum of the student model by adding a random uniform noise to the output of the teacher's hidden layer, i.e., $u_T(\mathbf{x}_T; \mathbf{\Theta}_T)$. We compared different range of noise from 0.05 to 0.5, and observed a slight improvement with a range of random uniform noise set to [-0.1, 0.1], i.e., the result of FRKD-P.

For the proposed method, i.e., ITSL, both Eq. 5 and Eq. 6 were used for optimization. For Eq. 5, we set λ to 0.3 by referring to the FRKD results. For Eq. 6, we set ξ to 0.1, and compared different γ with 0.1, 0.2 and 0.3. The best result was obtained when γ was set to 0.1. Compared with perturbation with a random noise, the proposed method obtained a more larger improvement. Compared with the DCNN model, the FRKD method obtained 23.1% relative improvement. The proposed method obtained 30.4% and 9.5% relative improvements than DCNN and FRKD methods, respectively.

Table 2. Investigation results of baseline systems and the proposed method with two-second utterances (UER %).

Methods	λ	γ	Valid.	Test
I-vector	-	-	-	11.11
GRU(256x2)	-	-	11.20	12.63
biGRU(256x2)	-	-	11.03	12.22
DCNN [14]	-	-	6.00	6.87
DCNN (4.0s Teacher)	-	-	2.43	2.83
FRKD [14]	0.1	-	4.83	5.67
FRKD [14]	0.3	-	4.17	**5.28**
FRKD [14]	0.5	-	4.23	5.33
FRKD [14]	0.7	-	4.74	5.49
FRKD-P (Perturbation)	0.3	-	4.43	5.17
ITSL (Proposed)	0.3	0.1	3.73	**4.78**
ITSL (Proposed)	0.3	0.2	4.03	5.12
ITSL (Proposed)	0.3	0.3	3.93	4.86

Table 3. Summary of the results of baseline, FRKD, FRKD with perturbation (FRKD-P) and ITSL (UER %).

Test	DCNN	FRKD	FRKD-P	ITSL
Test (2.0s)	6.87	5.28	5.17	**4.78**
Test (1.5s)	8.63	7.10	6.99	**6.67**
Test (1.0s)	13.18	12.12	11.94	**11.07**

We summarized the results of DCNN, FRKD, FRKD with random noise perturbation and the proposed method in Table 3. For all the student models, we used the same four-second-based teacher model. λ was set to 0.3 and γ, ξ were set to 0.1. From these results, we observed that the perturbation with random noise had a slight contribution for overcoming the local minimum problem. The proposed method showed its effectiveness on all the target duration utterances. For 2.0s, 1.5s, 1.0s test data, the proposed method obtained 9.5%, 6.1% and 8.7% relative improvements than the FRKD method, and obtained 30.4%, 22.7% and 16.0% relative improvements than the DCNN models. For LID tasks, the experiment results showed that the proposed method is an effective method for improving the performance on short utterances.

5. CONCLUSIONS

In this paper, we proposed an interactive teacher-student learning method for improving the optimization of student models for short utterance LID tasks. Different from conventional teacher-student frameworks use fixed pre-trained teacher models, the proposed method further tunes the teacher model with reference to the loss of the student models, that drives the optimization of student models to escape from local minimum. Experiment results showed that the proposed method is an effective method for short duration utterance LID tasks.

6. REFERENCES

[1] H. Li, B. Ma and K. A. Lee, "Spoken language recognition: From fundamentalsto practice," in Proc. of *The IEEE*, vol. 101, no. 5, pp. 1136-1159, 2013.

[2] C.-H. Lee, "Principles of spoken language recognition," in *Springer Handbook of Speech Processing and Speech Communication*, 2008.

[3] N. Dehak, P. Torres-Carrasquillo, D. Reynolds and R. Dehak, "Language recognition via ivectors and dimensionality reduction," in Proc. of *Interspeech*, 2011.

[4] S. Novoselov, T. Pekhovsky and K. Simonchik, "STC speaker recognition system for the NIST i-vector challenge," in Proc. of *Odyssey*, 2014.

[5] S. O. Sadjadi, J. W. Pelecanos and S. Ganapathy, "Nearest neighbor discriminant analysis for language recognition," in Proc. of *ICASSP*, 2015.

[6] Y. Song, B. Jiang, Y. Bao, S. Wei and L.-R. Dai, "I-vector representation based on bottleneck features for language identification," in *Electronics Letters*, vol. 49, no. 24, pp. 1569-1570, 2013.

[7] P. Shen, X. Lu, L. Liu and H. Kawai, "Local Fisher discrimiant analysis for spoken language identification," in Proc. of *ICASSP*, 2016.

[8] M. Najafian, S. Safavi, P. Weber and M. Russell, "Augmented Data Training of Joint Acoustic/Phonotactic DNN i-vectors for NIST LRE15,", in Proc. of *Odyssey*, 2016.

[9] I. Lopez-Moreno, J. Gonzalez-Dominguez, D. Martinez, O. Plchot, J. Gonzalez-Rodriguez and P. J. Moreno, "On the use of deep feedforward neural networks for automatic language identification," *Computer Speech & Language*, Vol.40, pp.46-59, 2016.

[10] I. Lopez-Moreno, J. Gonzalez-Dominguez, O. Plchot, D. Martinez, J. Gonzalez-Rodriguez and P. Moreno, "Automatic language identification using deep neural networks," in Proc. of *ICASSP*, 2014.

[11] A. Lozano-Diez, R. Zazo Candil, J. G. Dominguez, D. T. Toledano and J. G. Rodriguez, "An end-to-end approach to language identification in short utterances using convolutional neural networks," in Proc. of *Interspeech*, 2015.

[12] S. Fernando, V. Sethu, E. Ambikairajah and J. Epps, "Bidirectional Modelling for Short Duration Language Identification," in Proc. of *Interspeech*, 2017.

[13] W. Geng, W. Wang, Y. Zhao, X. Cai and B. Xu, "End-to-End Language Identification Using Attention-Based Recurrent Neural Networks," in Proc. of *Interspeech*, 2016.

[14] P. Shen, X. Lu, S. Li and H. Kawai, "Feature Representation of Short Utterances Based on Knowledge Distillation for Spoken Language Identification," in Proc. of *Interspeech*, 2018.

[15] G. Hinton, O. Vinyals and J. Dean, "Distilling the knowledge in a neural network," arXiv preprint arXiv:1503.02531, 2015.

[16] T. Asami and R. Masumura and Y. Yamaguchi and H. Masataki and Y. Aono, "Domain adaptation of DNN acoustic models using knowledge distillation," in Proc. of *ICASSP*, 2017.

[17] A. Romero, N. Ballas, S. E. Kahou, A. Chassang, C. Gatta and Y. Bengio, "Fitnets: Hints for thin deep nets," in Proc. of *ICLR*, 2015.

[18] D. Povey, et al., "The Kaldi speech recognition Toolkit," in Proc. of *ASRU*, 2011.

[19] J. Gonzalez-Dominguez, I. Lopez-Moreno, P. J. Moreno and J. Gonzalez-Rodriguez, "Frame by Frame Language Identification in Short Utterances using Deep Neural Networks," *Neural Networks Special Issue: Deep Learning of Representations*, Vol.64, pp.49-58, 2015.