

Mutual-learning Sequence-level Knowledge Distillation for Automatic Speech Recognition

Zerui Li^a, Yue Ming^{a,*}, Lei Yang^b and Jing-Hao Xue^c

^aBeijing Key Laboratory of Work Safety and Intelligent Monitoring, School of Electronic Engineering, Beijing University of Posts and Telecommunications, Beijing 100876, P.R. China.

^bCollege of Information Science and Engineering, Shandong Agricultural University, Tai'an 271018, P.R. China

^cDepartment of Statistical Science, University College London, London WC1E 6BT, U.K.

ARTICLE INFO

Keywords:

Automatic speech recognition (ASR)

Model compression

Knowledge distillation (KD)

Mutual learning

Connectionist temporal classification (CTC)

ABSTRACT

Automatic speech recognition (ASR) is a crucial technology for man-machine interaction. End-to-end models have been studied recently in deep learning for ASR. However, these models are not suitable for the practical application of ASR due to their large model sizes and computation costs. To address this issue, we propose a novel mutual-learning sequence-level knowledge distillation framework enjoying distinct student structures for ASR. Trained mutually and simultaneously, each student learns not only from the pre-trained teacher but also from its distinct peers, which can improve the generalization capability of the whole network, through making up for the insufficiency of each student and bridging the gap between each student and the teacher. Extensive experiments on the TIMIT and large LibriSpeech corpora show that, compared with the state-of-the-art methods, the proposed method achieves an excellent balance between recognition accuracy and model compression.

1. Introduction

With the advances of deep learning technology, end-to-end networks have significantly improved the performance of automatic speech recognition (ASR). The end-to-end ASR models based on recurrent neural network (RNN), such as connectionist temporal classification (CTC) [1, 2], recurrent neural network transducer (RNN-T) [3, 4] and attention-based model [5, 6], can bypass the label alignment stage to model from input acoustic features to output labels directly. However, the outstanding recognition accuracy of end-to-end models comes with a massive amount of parameters, computational costs and significantly redundant representations. Hence, end-to-end models are not suitable for practical ASR.

Model compression is a technique to reduce the model size with negligible accuracy loss. Several types of model compression approaches have been proposed, such as quantization [7, 8], matrix factorization [9, 10], pruning [11, 12] and knowledge distillation [13]. The knowledge distillation based methods can significantly reduce model sizes and computational costs, as well as combining with other compression methods to reduce footprint and runtime latency [14]. It uses a teacher-student framework to distill knowledge from a larger teacher model to guide a smaller student model. However, conventional knowledge distillation methods often lack the exploration of student models with distinct structures, which can provide distinct and complementary learning capabilities for ASR: a deep structure has a greater generalization capability and a wide structure is easier to build a long-term temporal dependence.

Therefore, in this paper, we propose a novel approach to ASR: mutual learning for sequence-level knowledge distil-

lation with distinct student structures. We focus on improving the abilities of student models through mutual learning among distinct students. Instead of training a single student model, we simultaneously train a set of student models with distinct structures. That is, students not only learn knowledge from the teacher model, but also learn from other students. On the one hand, due to structural differences, each student can get different knowledge transferred from the teacher network; on the other hand, mutual learning among students allows them to complement each other and make the best of knowledge from the teacher. Trained in this way, each student becomes more generalized than when learning alone or with all the same structure. To demonstrate the effectiveness of our proposed method, we shall conduct extensive experiments on the TIMIT and large LibriSpeech corpora.

Our main contributions are three-fold:

- We propose an approach to mutual-learning sequence-level knowledge distillation with distinct student structures, to train compact and accurate ASR networks, enabling students to learn from their peers and make up for their structural deficiencies.
- We extend our proposed approach to a multi-teacher knowledge distillation framework, which simultaneously transfers the knowledge of multiple teachers to different students.
- Through extensive experiments on the TIMIT and large scale LibriSpeech corpora, we demonstrate that our proposed method can significantly reduce the model size and computation cost with slight recognition accuracy decrease.

The rest of this paper is organized as follows: Section 2 reviews the related work, Section 3 revisits the CTC approach and knowledge distillation. Section 4 describes the proposed approach to mutual-learning sequence-level knowledge dis-

*Corresponding author

 myname35875235@126.com (Y. Ming)

ORCID(s):

tillation. The implementation details, experimental results and analysis are presented in Section 5, and the paper is concluded in Section 6.

2. Related Work

Model compression methods can be roughly divided into three types: neural architecture search (NAS), parameter compression and knowledge distillation.

NAS aims at automatically designing neural network architectures [15–18]. For example, He et al. [17] proposed the AutoML for Model Compression (AMC) method and introduced reinforcement learning to learn the optimal parameters of pruning; Dudziak et al. [18] used reinforcement learning to select the per-layer compression ratios based on matrix approximation. However, such a method requires massive computation during the search.

Parameter compression removes redundant information from complex trained models by, for example, pruning, quantization and matrix factorization. Takeda et al. [19] designed a score function to judge the importance of each node and proposed node-pruning to prune unimportant nodes. Dai et al. [20] proposed a hidden-layer LSTM and grow-and-prune training method to address the problems of model redundancy and runtime delay. Qian et al. [21] introduced binary neural network for acoustic modeling in speech recognition. Mori et al. [22] performed Tensor-Train decomposition on the weight matrix of the recurrent network to reduce the number of ASR parameters. Although these methods have achieved high parameter compression ratio with low-performance decrease, they require additional retraining.

Knowledge distillation was introduced by Hinton et al. [23] based on a teacher-student framework. The knowledge distillation based methods can transfer knowledge from a large teacher model to a small student model. Other compression techniques such as pruning and quantization can also be combined with knowledge distillation for further acceleration and compression. Our work is a knowledge distillation method, so here we focus on reviewing related knowledge distillation methods.

Romero et al. [24] extended knowledge distillation to intermediate representation and verified that the intermediate representation could improve the performance of the student model. Lee et al. [25] proposed a method to combine knowledge distillation and singular value decomposition with improving the quality of the knowledge transferred. Jiang et al. [26] proposed discriminant logit loss and category-aware attention loss to optimize the knowledge transferring process. Previous methods independently extract instance features from the teacher model as the distilled knowledge while ignoring the correlation between multiple instances. Liu et al. [27] introduced instance relationship graph for knowledge distillation to model not only instance features but also instance relationships as knowledge. Peng et al. [28] proposed correlation congruence for knowledge distillation to trans-

fer the correlation between instance features. Wu et al. [29] proposed a multi-teacher knowledge distillation framework to compress the model by transferring the knowledge from multiple teachers to a single student model. Wong et al. [30] proposed to train the student by an ensemble with a diversity of state cluster sets. Simultaneous distillation algorithms [31] trained simultaneously a set of models that learn from each other in a peer-teaching manner. Zhang et al. [32] proposed mutual learning to improve the performance of deep neural networks. Thoker et al. [33] proposed an approach that uses knowledge distillation for cross-modal action recognition with mutual learning to train a small ensemble of student networks.

Li et al. [34] trained a large-size DNN and used its output distribution to teach the small-size DNN by minimizing the Kullback–Leibler (KL) divergence of the output distribution between them in ASR. Kurata et al. [35] proposed to transfer the knowledge of the high-latency BiLSTM model to the low-latency UniLSTM model to improve the accuracy of streaming recognition. As such, output differences between teacher and student are minimized for each frame, called frame-level knowledge distillation [36]. Takashima et al. [37] found that the conventional knowledge distillation method based on frame-level cross-entropy made the performance of the student model worse. For standard ASR training, it is often found that the sequence-level training performs better than the framework-level training, and frame-level posteriors may not adequately convey information about the sequential nature of speech data. The sequence-level knowledge distillation has been proposed to ASR [38]. The sequence-level knowledge distillation uses the output of the teacher network to generate sequences with the k-best beam search and saves them as pseudo targets to train the student network. Takashima et al. [39] investigated the implementation of sequence-level knowledge distillation for CTC models and proposed a lattice-based sequence-level knowledge distillation method. Mun'im et al. [40] investigated the feasibility of sequence-level knowledge distillation for the sequence to sequence models. Moreover, penalizing teachers with higher WERs can reduce the accuracy of recognition results. Kim et al. [41] added an exponential weight to the sequence-level knowledge distillation method, which reflects the quality of the teacher model output by the weighing scheme to minimize the knowledge distillation loss function. Meng et al. [42] proposed conditional teacher-student framework, in which the student model selectively chooses to learn from either the ground truth labels or the outputs of the teacher model.

In contrast to the above methods, our method focuses on the structural differences between models in mutual learning. We propose an approach to mutual-learning sequence-level knowledge distillation with distinct student structures for ASR, aiming to learn more knowledge from the teacher by exploring the differences between the structures to increase recognition accuracy.

3. Knowledge Distillation for CTC-based Model

In this section, we first revisit the connectionist temporal classification (CTC) approach. Then, we introduce a general form of knowledge distillation and the sequence-level knowledge distillation in ASR.

3.1. Connectionist Temporal Classification

The CTC method is an objective function essentially for sequence labeling problems that brings significant benefits to the acoustic modeling of ASR. Generally, the length of input features is often longer than the length of output sequences in ASR. In order to deal with this problem in training, the key idea of CTC is to allow duplications of output labels and extend an additional blank symbol ϕ , which indicates that no labels are transmitted at a specific time step [43]. Thus, the CTC-based model can automatically infer the alignment of speech frames and labels, and does not require pre-alignments between input acoustic features and output characters. The CTC-based model addresses the problem of context-dependent state mismatch, making modeling simpler and easier. Moreover, the presence of a large number of blank symbols allows the model to use frame skipping during the decoding process, thus considerably speeding up the decoding process.

In the CTC framework, the intermediate label representation $\pi = (\pi_1, \dots, \pi_T)$ (called a path) is converted into output sequence by deleting repeated and blank labels. CTC trains the model to maximize the sum of the probabilities of all possible paths [44]:

$$p(y|x) = \sum_{\pi \in B(y)} p(\pi|x), \quad (1)$$

where y denotes the label sequence, x is the input sequence and $B(y)$ is the set of CTC paths for label sequence y . The probability of label sequence $p(\pi|x)$ is calculated as

$$p(\pi|x) = \prod_{t=1}^T p(\pi_t|x), \quad (2)$$

where $p(\pi_t|x)$ is the posterior probability of label π_t at time t given the input x .

Then the loss function L_{CTC} of CTC is defined as the sum of negative log probability of correct labelings for each training sequence:

$$L_{CTC} = - \sum_{x,y} \ln p(y|x). \quad (3)$$

The model is trained by minimizing L_{CTC} .

3.2. Knowledge Distillation

Knowledge distillation is a model compression method for deep neural networks. The main idea of knowledge distillation is to use the soft target obtained through a well-trained teacher model to guide the training of the student model. In the knowledge distillation framework, the teacher model is trained by the ground truth labels. Then the student model is

trained by both the soft targets from the well-trained teacher model and the ground truth labels, by using the following loss function:

$$L_{KD}(\theta) = \alpha L_{CTC}(\theta) + (1 - \alpha)L_{KLD}(\theta), \quad (4)$$

where L_{CTC} is the CTC loss, L_{KLD} is the KL divergence, α is the hyper-parameters to balance L_{CTC} and L_{KLD} , and θ denotes the student model. The KL divergence of the student output distribution to the teacher output distribution can be formulated as

$$L_{KLD} = \sum P(x) \log(P(x)/Q(x)), \quad (5)$$

where $Q(x)$ is the output distribution of the student model and $P(x)$ is the output distribution of the teacher model. Because $P(x) \log P(x)$ is constant as the teacher model is fixed, minimizing the loss function is equivalent to minimizing the following equation:

$$L_{KLD} = - \sum P(x) \log Q(x). \quad (6)$$

The output distribution $P(x)$ is represented as a softmax probability with the temperature index.

3.3. Sequence-level Knowledge Distillation

In the previous work of knowledge distillation, Eq.(4) is calculated for each frame, which we call the frame-level knowledge distillation. The frame-level information propagated in speech recognition may not effectively capture the behaviors of the teacher at the sequence level. Instead, it may be better for the student to learn from the teacher's sequence-level behaviors directly. Thus, a sequence-level knowledge distillation has been used in ASR [38], which uses the output of the teacher model to generate sequences with the k-best beam search and saves them as pseudo targets to train the student network model.

Following [39], we use the output sequences from teacher model as pseudo labels to train student models. The pseudo targets are similar to soft targets in the frame-level knowledge distillation. Even if there are some errors in pseudo targets, the student is supposed to achieve better performance with sequence-level knowledge distillation, because the student tries to imitate the distribution of teacher instead of modeling the distribution of training data directly. Then, using the pseudo targets, we train a student model under sequence-level knowledge distillation [40]:

$$L_{seqKD} = - \sum_{t \in \mathcal{T}} \mathbb{1}\{t = \hat{y}\} \log p(t|x) = - \log p(t = \hat{y}|x), \quad (7)$$

where x is the input sequence, \mathcal{T} denotes an approximation of the all possible sequences, $\mathbb{1}\{\}$ is the indicator function and \hat{y} is the output hypothesis estimated by the teacher model.

4. Mutual-learning Sequence-level Knowledge Distillation

In this section, we first propose our framework, the mutual-learning sequence-level knowledge distillation, and then extend it to a multi-teacher knowledge distillation framework.

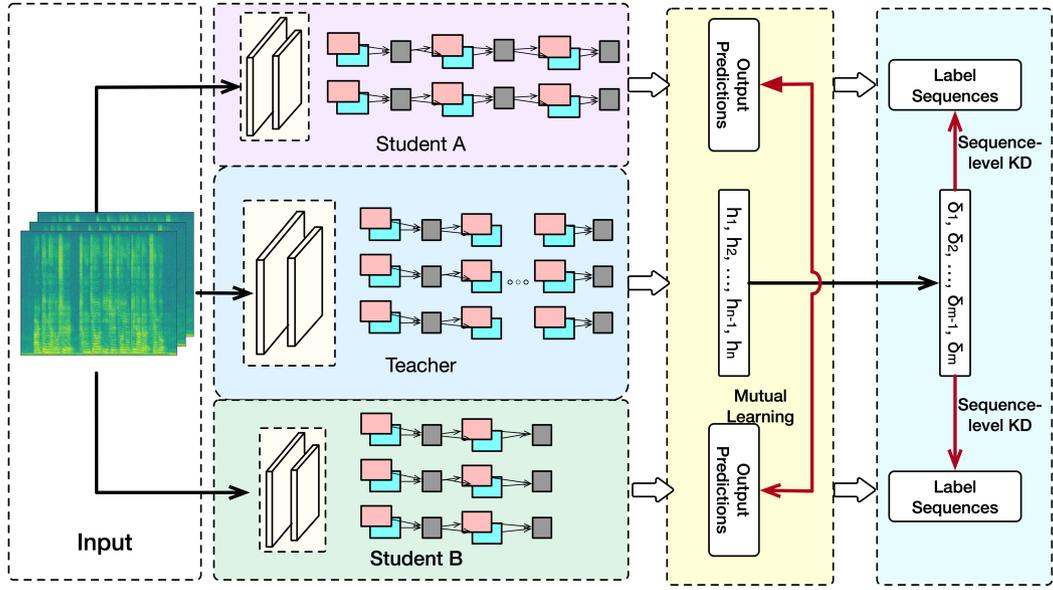


Figure 1: Mutual-learning sequence-level knowledge distillation with distinct structures. The framework consists of three models: Teacher A, Student A and Student B. Student A has a deeper network, and Student B has a wider network. The output predictions are denoted by h and hypotheses labels are denoted by δ .

4.1. Mutual Learning for Knowledge Distillation

The original knowledge distillation method is to train a single small student model by using the soft targets obtained from a well-trained teacher model. In contrast, in mutual learning, a group of students can learn through collaboration and the students learn from each other throughout the training process to solve the task together.

Deep neural network training is a non-convex problem and there may be local optimal value problems. Because of this, the final trained model is often sensitive to the parameter initialization. Starting from different random parameter initializations, it is possible that the models may converge to different local optima after training and therefore may show diversity in their outputs. It is this different initialization information that provides additional information in mutual learning for knowledge distillation.

In mutual learning, the students usefully pool their collective estimates of the next most possible labels. According to their partner, finding out and matching the most possible labels for each input feature increases the posterior probability of each student. Mutual learning provides a simple and efficient method to improve the generalization capability of the model through collaborative learning with other models. The student models trained with mutual learning outperform the students who are individually trained by a single sizeable well-trained teacher.

4.2. Mutual Learning with Distinct Structures

In our proposed framework of mutual-learning sequence-level knowledge distillation, we choose student models with distinct structures because the learning capabilities of different structure models are different. One student model is deeper and the other model is wider. The deeper structure can improve network expressiveness and correct some

wrong samples, helping the shallow network learn easier. On the contrary, the wider structure can more effectively obtain contextual relationships of time series, helping the deep network learn the long-term dependence. The collaboration of these distinct student models would introduce more diversity in the students' outputs and achieve a better performance of ASR by using multiple forms of diversities.

Therefore, we focus on the differences in student network structures and aim to overcome each student's drawback in structure with mutual learning. Similar to the different initializations, since each student has a different network structure, their probability estimates for the most likely labels will differ. If all the students make the errors in their outputs, then little can be gained from mutual learning; in contrast, if the models make different errors in their outputs, the students may be able to correct their errors through mutual learning. That is, the mutual learning among students can be regarded as a learning group, where the student models complement each other by exploring the structural differences between each other to obtain better performance. Thus, each student can make the best of rich and correct knowledge transferred by the teacher, while avoiding receiving incorrect knowledge, through learning and interactions with other students.

The overall framework of the proposed method is illustrated in Figure 1. Specifically, each student (e.g. $stu1$) is trained with three losses: a sequence-level knowledge distillation loss between teacher and student L_{seqKD} , a CTC loss L_{CTC} and a mimicry loss $L_{KL-stu2}$ that aligns class posterior of the student $stu1$ with the class probabilities of the other student $stu2$. This can be expressed as

$$L_{stu1} = (1 - \alpha)L_{CTC} + \alpha(L_{seqKD} + \beta L_{KL-stu2}), \quad (8)$$

where L_{CTC} denotes the CTC loss function, $L_{KL-stu2}$ is the KL divergence between the two student networks, and β is

the weight of the KL divergence.

4.3. Extension to Multi-teacher Framework

As group knowledge from a group of teachers can also be compressed into a single student by using teacher-student learning, to further improve the recognition accuracy, we extend the proposed approach to multiple teachers, leading to a multi-teacher mutual-learning sequence-level knowledge distillation framework for ASR. The extended loss function for the j th student $stuj$ becomes

$$L_{stuj} = (1-\alpha)L_{CTC} + \alpha \left(\sum_{tea} L_{seqKD} + \beta \sum_{i \neq j} L_{KL-stui} \right), \quad (9)$$

where tea denotes the group of teachers. In this way, a group of student networks learns knowledge from the output probability distributions of their peers and a group of the teachers, as well as the ground truth labels.

5. Experimental Studies

In this section, we first introduce the evaluation datasets and models. Then we evaluate our proposed approach on the TIMIT and LibriSpeech corpuses and compare its performance with state-of-the-art methods.

5.1. Experiments Setup

5.1.1. Datasets and setup

The speech recognition experiments were conducted on the TIMIT and LibriSpeech datasets. The TIMIT corpus is the mainstream datasets in ASR experiments. It contains 6,300 sentences in total, including 630 speakers, and the total time is five hours. The training set is about 70% of the total, and the test set is approximately 30%. At the same time, we used the 1,000-hours LibriSpeech corpus to verify our proposed method on large data sets. LibriSpeech is a large corpus developed for ASR. It consists of 1000 hours of reading English speech, based on public domain audiobooks of the LibriVox project. The training set is about 95% of the total and the rest dataset is for validation and test. In LibriSpeech, the validation data and test data are split into ‘clean’ and ‘other’ subsets.

Commonly used features in speech include LSF [45–47], LPC [48], MFCC [49], FBank [50] and spectrogram. We sampled all audio data at 16kHz. In our experiment, all the models were trained on the log spectrogram features. The input features can be depicted as 2-dimensional spectrograms with time and frequency axes, extracted from a 25ms window and shifted every 10ms. The dimension on the frequency axes is 201 dimensions and the dimension on the time axes depends on the input sequence length. We applied standard mean and deviation normalization to audio tensor. We select training samples longer than 1 second and shorter than 15 seconds as the model input. We chose 29 distinct characters (including 26 capital English letters, space, single quote character and blank token) as model units. We used Stochastic Gradient Descent (SGD) with mini-batch of 32, and the learning rate started at 0.0003. At the end of

each epoch, the learning rate was multiplied by a factor of 0.95. The training would be continued until the recognition error on the development set increased, in order to select the best model. For decoding, the greedy decoding procedure (no complex beam search decoder or external language model) was used. This makes our end-to-end ASR systems all-neural.

5.1.2. Model structures

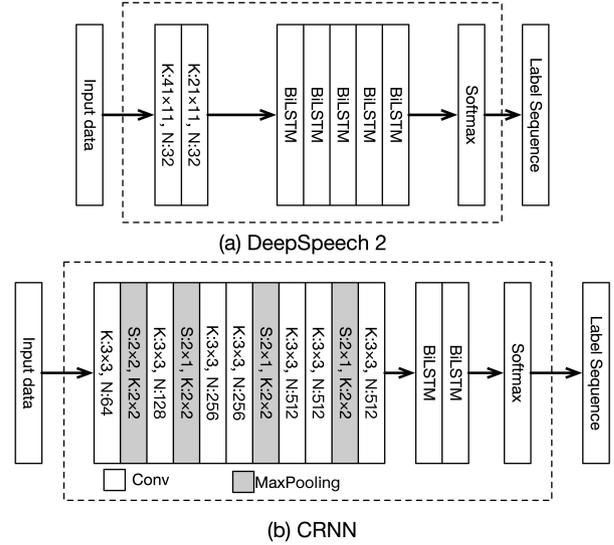


Figure 2: Model Structure. We choose DeepSpeech2 and CRNN as our teacher models.

We adopt DeepSpeech2 [2] and CRNN [51] as Teacher A and Teacher B, respectively. The DeepSpeech2 and CRNN based on the Connectionist Temporal Classification (CTC) criterion are natural end-to-end (E2E) systems directly targeting word as the output unit. We use spectrogram as the input feature, and it is necessary to extract higher-level feature representations using CNNs. As shown in Figure 2, the DeepSpeech2 network has two convolution layers and five layers of bidirectional long short-term memory (BLSTM), with 1024 hidden units in each layer; the CRNN consists of three parts: seven convolutional layers, two recurrent layers, and one transcription layer, from bottom to top.

For student networks, we choose two differently-structured LSTM networks, one student model is deeper and the other is wider. The divergence between the two models will provide more extra information for mutual-learning sequence-level knowledge distillation. One has two LSTM hidden layers with 512 hidden units for each layer, and the other has five LSTM hidden layers with 256 hidden units for each layer. The output layer of all models is linearly projected to 29 dimensions.

As shown in Table 1, the numbers of model parameters of Teacher A, Teacher B, Student A, and Student B models are 58.3 million, 54.5 million, 11.9 million and 7.8 million, respectively. We can see that the student model can achieve more than five times the compression of the teacher model.

Table 1

The number of parameters of teacher and student models

	Model parameters
Teacher LSTM	58.3M
Teacher CRNN	54.5M
Student A	11.9M
Student B	7.8M

Table 2

Results (CER) from training with mutual learning for knowledge distillation on TIMIT

	Size	CER
Teacher A	-	20.348
Student A	20.4%	28.393
Student A + seqKD	20.4%	25.836
A+MLKD(with Student B)	20.4%	24.757
Student B	13.4%	27.818
Student B + seqKD	13.4%	26.064
B+MLKD(with Student A)	13.4%	24.733

5.2. Results and Analysis

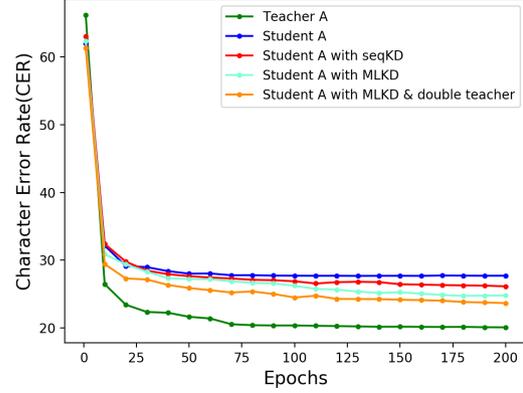
5.2.1. Mutual-learning sequence-level knowledge distillation with distinct structures (MLKD)

We first validate the performance of the proposed method on TIMIT. The results in terms of character error rate (CER) are shown in Table 2, where Teacher A, Student A and Student B are trained from scratch; Student A + seqKD means that Student A learns from Teacher A in a sequence-level knowledge distillation way; A+MLKD(with Student B) indicates that Student A and Student B learn in the proposed mutual-learning sequence-level knowledge distillation way. Similarly, for the block of Student B in Table 2.

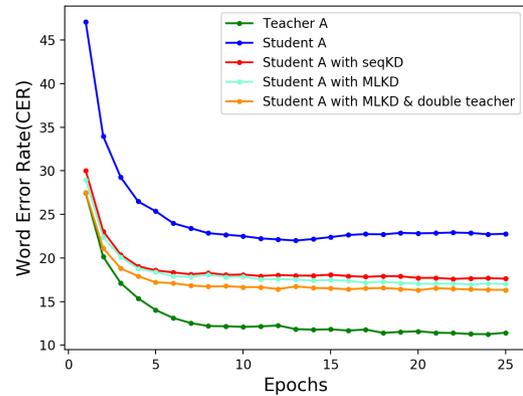
We can observe the following patterns from Table 2. First, the size of student models and knowledge distillation models are much smaller than the teacher model; for example, for Student A, the number of parameters is only about 20% of Teacher A. Secondly, with the knowledge distillation training, the student models (e.g. Student A + seqKD) can achieve better CER than it training directly without knowledge distillation (e.g. Student A). Thirdly, because of the benefit from using mutual learning with distinct student models, the proposed model (e.g. A+MLKD(with Student B)) achieved the best CER (24.757%).

Moreover, we show in Figure 3 the performance of different models on the TIMIT corpus, from which we can also see that the mutual learning models (Student A with MLKD) outperform individual student models (Student A and Student A with seqKD).

We further evaluate the models on the LibriSpeech corpus under four scenarios: dev-clean, dev-other, test-clean and test-other. In this work, we have not used a separate language model for further reduction of word error rate (WER). Table 3 shows the WER of the teacher model and the student models. The observations from Table 3 are generally consistent with those of Table 2. In particular, the WER of the

**Figure 3: The performance, in terms of character error rate (CER), of different methods on TIMIT**

proposed mutual learning models result in remarkable improvement in WER: for example, A+MLKD(with Student B) on dev-clean is 15.862%, a 4.854% reduction of the Student A baseline (20.716%); the WER of 13.829% on dev-clean for Student B was achieved for our proposed method B+MLKD(with Student A), yielding a reduction of 4.442% over the Student B baseline (18.271%). The same pattern can be seen from Figure 4 that the mutual learning models (Student A with MLKD) outperform individual student models (Student A and Student A with seqKD).

**Figure 4: The performance, in terms of word error rate (WER), of different methods on LibriSpeech**

Mutual learning improves upon the single student performances over single conventional training. The results may suggest that mutual learning knowledge distillation slightly increases the ensemble diversity and improves the learning ability. Another benefit of the ability to learn from different models is to make up for structural defects. The deep model can learn to capture longer span temporal dependencies in the data and the wide model can learn generalization ability from the deep model. The Student B performs better than Student A with much less number of parameters. This may be because Student B has a deeper network structure and

Table 3

Results (WER) from training with mutual learning for knowledge distillation on LibriSpeech

	Size	dev-clean	dev-other	test-clean	test-other
Teacher A	-	10.285	28.015	10.716	28.896
Student A	20.4%	20.716	42.930	20.701	44.705
Student A + seqKD	20.4%	17.578	38.745	17.744	39.684
A+MLKD(with Student B)	20.4%	15.862	37.065	16.222	38.557
Student B	13.4%	18.271	38.826	18.050	40.124
Student B + seqKD	20.4%	16.748	36.832	16.500	38.040
B+MLKD(with Student A)	13.4%	13.829	32.667	13.655	34.020

Table 4

Results (CER) of multi-teacher MLKD on TIMIT

	Size	CER
Teacher A	-	20.348
Teacher B	-	20.104
Student A + Mutli-teacher	20.4%	25.103
A+MLKD(with Student B)	20.4%	24.757
A+MLKD(with Student B & Mutli-teacher)	20.4%	23.685
Student B + Mutli-teacher	13.4%	24.984
B+MLKD(with Student A)	13.4%	24.733
B+MLKD(with Student A & Mutli-teacher)	13.4%	22.496

better generalization ability. With the help of Student A, the ability to capture longer time span dependencies has been further improved. The experiments show that mutual learning with different structures helps students learn more from the teacher than the conventional knowledge distillation.

5.2.2. Multi-teacher MLKD framework

In the previous experiments, the students only learn from a single teacher. Fukuda et al. [52] leverage information from multiple teachers by training student networks with a group of teachers. As presented in section 4.3, we extend our proposed method, the mutual-learning sequence-level knowledge distillation with distinct structures (MLKD), to a multi-teacher MLKD framework.

The evaluation procedure and results of various combinations of student models and teacher models on the TIMIT corpus are listed in Table 4.

We first train a single Student A with multiple teachers (Student A + Multi-teacher), the result of which (with CER of 25.103%) show that leveraging the output of multiple teachers for training student models is better than training with a single teacher (Student A + seqKD with CER of 25.836% as shown in Table 2).

Then we further illustrate the efficacy of our proposed mutual learning knowledge framework: compared with training with multiple teachers, Student A+MLKD (with Student B) leads to a 0.346% CER reduction (25.103% to 24.757%) from Student A + Multi-teacher; Student B+MLKD (with Student A) also achieves a 0.251% CER reduction (24.984% to 24.733%) from Student B + Multi-teacher. The patterns on the LibriSpeech dataset are consistent with the patterns on the TIMIT (as shown in the Table 5). The performance

of distinct students with mutual learning is more competitive than a single student with multiple teachers.

Finally, as we can from Table 4, MLKD combined with two teachers produces the best performance: Student A with multi-teacher MLKD (i.e. A+MLKD (with Student B & Multi-teacher)) achieves 23.685% CER; Student B with multi-teacher MLKD also achieves the best performance at 22.496% CER. This pattern can also be clearly observed in Figure 3 on the TIMIT corpus.

As shown in the Table 5 and Figure 4, a similar pattern can be observed for Student A to support multi-teacher MLKD on the LibriSpeech corpus. However, this pattern is only observed for Student B on the dev-clean case but not the other three scenarios. This may be because the WER of Student B with MLKD is already close to that of Teacher B.

5.2.3. Computation cost

The purpose of our method is to use small models to save storage space and computational consumption. We save more than half storage space by reducing 80% of parameters and boost computation by decreasing about 81% FLOPs compared with Teacher A. The proposed method has achieved the notable result on Student B, by removing 86.6% parameters and 84.4% FLOPs compared with Teacher A. That is, our methods reduce both the FLOPs and the memory consumption of the network, while maintaining high recognition accuracy rate.

5.2.4. Comparison with state-of-the-art methods

The comparison results on TIMIT and LibriSpeech of our proposed method with state-of-the-art methods are shown in Table 7. Takashima et al. [37] investigated sequence-level

Table 5
Results (WER) of multi-teacher MLKD on LibriSpeech

	Size	dev-clean	dev-other	test-clean	test-other
Teacher A	-	10.285	28.015	10.716	28.896
Teacher B	-	13.365	31.870	13.468	32.658
Student A + Mutli-teacher	20.4%	17.062	38.227	16.713	39.440
A+MLKD(with Student B)	20.4%	15.862	37.065	16.222	38.557
A+MLKD (with Student B & Mutli-teacher)	20.4%	15.152	35.752	15.106	37.237
Student B + Mutli-teacher	20.4%	15.049	35.052	14.952	36.455
B+MLKD(with Student A)	13.4%	13.829	32.667	13.655	34.020
B+MLKD(with Student A & Mutli-teacher)	13.4%	13.810	33.199	13.727	34.287

Table 6
Computation costs

Model	Parameters	FLOPs	The reduction of FLOPs compared with Teacher A
DeepSpeech	58.3M	16.49G	-
CRNN	54.5M	22.19G	-
Student A	11.9M	3.11G	81%
Student B	7.8M	2.57G	84.4%

Table 7
Evaluation results on TIMIT and LibriSpeech compared with the state-of-the-art methods

Method	CER on TIMIT	WER on LibriSpeech
Sequence-level KD [37]	25.836	17.578
Essence KD [53]	25.890	17.163
ERR-KD [41]	25.869	17.251
Our MLKD	24.757	15.862
Our MLKD + multi-teacher	23.685	15.152

knowledge distillation to train a CTC-based model. Essence knowledge distillation [53] only selected the top k values (the essence knowledge) from the teacher model. Kim et al. [41] proposed to add an exponential weight coefficient to the sequence-level knowledge distillation method to balance the recognition quality of the teacher model. We chose the DeepSpeech2 as the teacher model, selected two layers of LSTM as the student model, and experimented them with the Sequence-level KD, Essence KD and ERR-KD. We carried out speech recognition experiments on the methods mentioned above and our MLKD method. As can be seen from Table 7, our MLKD is superior to these methods, remarkably improving the error rate on both TIMIT and LibriSpeech, which benefits from mutual learning from other students. Therefore, we can conclude that we obtain the competitive performance, proving the effectiveness of our proposed method. Furthermore, the performance of students trained with our proposed method is about the same as their teacher model.

5.2.5. Ablation study

Here we an ablation study to prove the effectiveness of using distinct student structures by comparing it with the

Table 8
Results (CER) of ablation study on TIMIT

	Size	CER
A+MLKD(with Student A)	20.4%	25.217
A+MLKD(with Student B)	20.4%	24.757
B+MLKD(with Student B)	13.4%	25.086
B+MLKD(with Student A)	13.4%	24.733

Table 9
Results (WER) of ablation study on LibriSpeech

	Size	dev-clean	dev-other	test-clean	test-other
A+MLKD(with Student A)	20.4%	17.135	37.723	16.998	39.190
A+MLKD(with Student B)	20.4%	15.862	37.065	16.222	38.557
B+MLKD(with Student B)	13.4%	15.979	36.153	16.089	37.214
B+MLKD(with Student A)	13.4%	13.829	32.667	13.655	34.020

same student structure. The results are shown in Table 8 and Table 9. On TIMIT, compared with the mutual learning of the same structure, MLKD with different student structures can reduce CER by 0.46% (from 25.217% to 24.757%) for Student A and by 0.353% (from 25.086% to 24.733%) for Student B. The results of LibriSpeech shown in Table 9 also suggest the same pattern: the mutual learning knowledge distillation with different student structures is superior to that with the same student structure.

6. Conclusion

In this paper, we investigate the mutual-learning sequence-level knowledge distillation framework with distinct student structures in automatic speech recognition. By exploring structural differences, the students can make up for the shortcomings of their structures and learn better from the teacher. Experiments on the TIMIT and LibriSpeech corpuses demonstrate that the proposed mutual learning method can produce a remarkable performance improvement, compared with the student models learning alone. Moreover, the extension of the proposed method to the multi-teacher framework can generally further improve the performance. Moreover, compared with the large teacher model, the proposed model reduces a significant amount in model size and manages no much decrease in recognition accuracy.

We choose sentences longer than 1 second and shorter

than 15 seconds to train and validate the models. In future work, to solve the mismatch between training with short-form audio and inference for long-form audio, we will investigate and evaluate the performance of various models under long-form audio.

Acknowledgements

The work presented in this paper was partly supported by Natural Science Foundation of China (Grant No. 62076030), Beijing Natural Science Foundation of China (Grant No. L182033) and the Fundamental Research Funds for the Central Universities (2019PTB-001).

References

- [1] A. Graves, S. Fernández, F. Gomez, J. Schmidhuber, Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks, in: ICML, ACM, 2006, pp. 369–376.
- [2] D. Amodei, S. Ananthanarayanan, R. Anubhai, J. Bai, E. Battenberg, C. Case, J. Casper, B. Catanzaro, Q. Cheng, G. Chen, et al., Deep Speech 2: End-to-end speech recognition in English and Mandarin, in: ICML, 2016, pp. 173–182.
- [3] T. N. Sainath, C.-C. Chiu, R. Prabhavalkar, A. Kannan, Y. Wu, P. Nguyen, Z. Chen, Improving the performance of online neural transducer models, in: ICASSP, IEEE, 2018, pp. 5864–5868.
- [4] Y. He, T. N. Sainath, R. Prabhavalkar, I. McGraw, R. Alvarez, D. Zhao, D. Rybach, A. Kannan, Y. Wu, R. Pang, et al., Streaming end-to-end speech recognition for mobile devices, in: ICASSP, IEEE, 2019, pp. 6381–6385.
- [5] W. Chan, N. Jaitly, Q. Le, O. Vinyals, Listen, attend and spell: A neural network for large vocabulary conversational speech recognition, in: ICASSP, IEEE, 2016, pp. 4960–4964.
- [6] L. Dong, S. Xu, B. Xu, Speech-transformer: a no-recurrence sequence-to-sequence model for speech recognition, in: ICASSP, IEEE, 2018, pp. 5884–5888.
- [7] N. Floropoulos, A. Tefas, Complete vector quantization of feedforward neural networks, *Neurocomputing* 367 (2019) 55–63.
- [8] F. Tung, G. Mori, Deep neural network compression by in-parallel pruning-quantization, *IEEE transactions on pattern analysis and machine intelligence* 42 (3) (2020) 568–579.
- [9] S. Lin, R. Ji, C. Chen, D. Tao, J. Luo, Holistic CNN compression via low-rank decomposition with knowledge transfer, *IEEE transactions on pattern analysis and machine intelligence* 41 (12) (2019) 2889–2905.
- [10] S. Swaminathan, D. Garg, R. Kannan, F. Andres, Sparse low rank factorization for deep neural network compression, *Neurocomputing* (2020).
- [11] S. Chen, Q. Zhao, Shallowing deep networks: Layer-wise pruning based on feature representations, *IEEE transactions on pattern analysis and machine intelligence* 41 (12) (2019) 3048–3056.
- [12] Z. Xie, L. Zhu, L. Zhao, B. Tao, L. Liu, W. Tao, Localization-aware channel pruning for object detection, *Neurocomputing* (2020).
- [13] M. Yuan, Y. Peng, CKD: Cross-task knowledge distillation for text-to-image synthesis, *IEEE Transactions on Multimedia* (2019).
- [14] H. Ding, K. Chen, Q. Huo, Compressing CNN-DBLSTM models for OCR with teacher-student learning and tucker decomposition, *Pattern Recognition* 96 (2019) 106957.
- [15] M. Tan, B. Chen, R. Pang, V. Vasudevan, M. Sandler, A. Howard, Q. V. Le, MnasNet: Platform-aware neural architecture search for mobile, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 2820–2828.
- [16] L. Jia, W. Feng, C. Chen, J. Zhang, Neural architecture search based on model pool for wildlife identification, *Neurocomputing* (2020).
- [17] Y. He, J. Lin, Z. Liu, H. Wang, L.-J. Li, S. Han, AMC: AutoML for model compression and acceleration on mobile devices, in: Proceedings of the European Conference on Computer Vision, 2018, pp. 784–800.
- [18] Ł. Dudziak, M. S. Abdelfattah, R. Vipperla, S. Laskaridis, N. D. Lane, ShrinkML: End-to-end ASR model compression using reinforcement learning, in: Interspeech, 2019, pp. 2235–2239.
- [19] R. Takeda, K. Nakadai, K. Komatani, Node pruning based on entropy of weights and node activity for small-footprint acoustic model based on deep neural networks., in: Interspeech, 2017, pp. 1636–1640.
- [20] X. Dai, H. Yin, N. Jha, Grow and prune compact, fast, and accurate LSTMs, *IEEE Transactions on Computers* 69 (3) (2020) 441–452.
- [21] Y.-m. Qian, X. Xiang, Binary neural networks for speech recognition, *Frontiers of Information Technology & Electronic Engineering* 20 (5) (2019) 701–715.
- [22] T. Mori, A. Tjandra, S. Sakti, S. Nakamura, Compressing end-to-end ASR networks by tensor-train decomposition., in: Interspeech, 2018, pp. 806–810.
- [23] G. Hinton, O. Vinyals, J. Dean, Distilling the knowledge in a neural network, *arXiv preprint arXiv:1503.02531* (2015).
- [24] A. Romero, N. Ballas, S. E. Kahou, A. Chassang, C. Gatta, Y. Bengio, FitNets: Hints for thin deep nets, *arXiv preprint arXiv:1412.6550* (2014).
- [25] S. H. Lee, D. H. Kim, B. C. Song, Self-supervised knowledge distillation using singular value decomposition, in: European Conference on Computer Vision, Springer, 2018, pp. 339–354.
- [26] L. Jiang, W. Zhou, H. Li, Knowledge distillation with category-aware attention and discriminant logit losses, in: ICME, IEEE, 2019, pp. 1792–1797.
- [27] Y. Liu, J. Cao, B. Li, C. Yuan, W. Hu, Y. Li, Y. Duan, Knowledge distillation via instance relationship graph, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 7096–7104.
- [28] B. Peng, X. Jin, J. Liu, D. Li, Y. Wu, Y. Liu, S. Zhou, Z. Zhang, Correlation congruence for knowledge distillation, in: Proceedings of the IEEE International Conference on Computer Vision, 2019, pp. 5007–5016.
- [29] M.-C. Wu, C.-T. Chiu, K.-H. Wu, Multi-teacher knowledge distillation for compressed video action recognition on deep neural networks, in: ICASSP, IEEE, 2019, pp. 2202–2206.
- [30] J. H. M. Wong, M. J. F. Gales, Y. Wang, General sequence teacher-student learning, *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 27 (11) (2019) 1725–1736.
- [31] X. Lan, X. Zhu, S. Gong, Knowledge distillation by on-the-fly native ensemble, in: Neural Information Processing Systems, Curran Associates Inc, 2018, pp. 7528–7538.
- [32] Y. Zhang, T. Xiang, T. M. Hospedales, H. Lu, Deep mutual learning, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 4320–4328.
- [33] F. M. Thoker, J. Gall, Cross-modal knowledge distillation for action recognition, in: ICIP, IEEE, 2019, pp. 6–10.
- [34] J. Li, R. Zhao, J.-T. Huang, Y. Gong, Learning small-size DNN with output-distribution-based criteria, in: Interspeech, 2014, pp. 1910–1914.
- [35] G. Kurata, K. Audhkhasi, Improved knowledge distillation from bi-directional to uni-directional LSTM CTC for end-to-end speech recognition, in: IEEE Spoken Language Technology Workshop (SLT), IEEE, 2018, pp. 411–417.
- [36] L. Lu, M. Guo, S. Renals, Knowledge distillation for small-footprint highway networks, in: ICASSP, IEEE, 2017, pp. 4820–4824.
- [37] R. Takashima, S. Li, H. Kawai, An investigation of a knowledge distillation method for CTC acoustic models, in: ICASSP, IEEE, 2018, pp. 5809–5813.
- [38] M. Huang, Y. You, Z. Chen, Y. Qian, K. Yu, Knowledge distillation for sequence model, in: Interspeech, 2018, pp. 3703–3707.
- [39] R. Takashima, L. Sheng, H. Kawai, Investigation of sequence-level knowledge distillation methods for CTC acoustic models, in: ICASSP, IEEE, 2019, pp. 6156–6160.
- [40] R. M. Mun'im, N. Inoue, K. Shinoda, Sequence-level knowledge distillation for model compression of attention-based sequence-to-

- sequence speech recognition, in: ICASSP, IEEE, 2019, pp. 6151–6155.
- [41] H.-G. Kim, H. Na, H. Lee, J. Lee, T. G. Kang, M.-J. Lee, Y. S. Choi, Knowledge distillation using output errors for self-attention end-to-end models, in: ICASSP, IEEE, 2019, pp. 6181–6185.
- [42] Z. Meng, J. Li, Y. Zhao, Y. Gong, Conditional teacher-student learning, in: ICASSP, IEEE, 2019, pp. 6445–6449.
- [43] S. Tong, P. N. Garner, H. Bourlard, Cross-lingual adaptation of a CTC-based multilingual acoustic model, *Speech Communication* 104 (2018) 39–46.
- [44] Y. Shi, M.-Y. Hwang, X. Lei, End-to-end speech recognition using a high rank LSTM-CTC based model, in: ICASSP, IEEE, 2019, pp. 7080–7084.
- [45] Z. Ma, A. Leijon, optimized LSF vector quantization based on beta mixture models, in: Eleventh Annual Conference of the International Speech Communication Association, 2010.
- [46] Z. Ma, A. Leijon, W. B. Kleijn, Vector quantization of LSF parameters with a mixture of Dirichlet distributions, *IEEE Transactions on Audio, Speech, and Language Processing* 21 (9) (2013) 1777–1790.
- [47] Z. Ma, S. Chatterjee, W. B. Kleijn, J. Guo, Dirichlet mixture modeling to estimate an empirical lower bound for lsf quantization, *Signal Processing* 104 (2014) 291–295.
- [48] N. Dave, Feature extraction methods LPC, PLP and MFCC in speech recognition, *International Journal for Advance Research in Engineering and Technology* 1 (6) (2013) 1–4.
- [49] N. Zeghidour, N. Usunier, I. Kokkinos, T. Schaiz, G. Synnaeve, E. Dupoux, Learning filterbanks from raw speech for phone recognition, in: ICASSP, IEEE, 2018, pp. 5509–5513.
- [50] J. Hai, E. M. Joo, Improved linear predictive coding method for speech recognition, in: Fourth International Conference on Information, Communications and Signal Processing, 2003 and the Fourth Pacific Rim Conference on Multimedia. Proceedings of the 2003 Joint, Vol. 3, IEEE, 2003, pp. 1614–1618.
- [51] B. Shi, X. Bai, C. Yao, An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39 (11) (2016) 2298–2304.
- [52] T. Fukuda, M. Suzuki, G. Kurata, S. Thomas, J. Cui, B. Ramabhadran, Efficient knowledge distillation from an ensemble of teachers, in: Interspeech, 2017, pp. 3697–3701.
- [53] Z. Yang, C. Zhang, W. Zhang, J. Jin, D. Chen, Essence knowledge distillation for speech recognition, arXiv preprint arXiv:1906.10834 (2019).