Adversarial Multi-Teacher Distillation for Semi-Supervised Relation Extraction

Wanli Li, *Student Member, IEEE*, Tieyun Qian, *Member, IEEE*, Xuhui Li, and Lixin Zou

***Abstract*— The shortage of labeled data has been a long-standing challenge for relation extraction (RE) tasks. Semi-supervised RE (SSRE) is a promising way through annotating unlabeled samples with pseudolabels as additional training data. However, some pseudolabels on unlabeled data might be erroneous and will bring misleading knowledge into SSRE models. For this reason, we propose a novel adversarial multi-teacher distillation (AMTD) framework, which includes *multi-teacher knowledge distillation* and *adversarial training* (AT), to capture the knowledge on unlabeled data in a refined way. Specifically, we first develop a general knowledge distillation (KD) technique to learn not only from pseudolabels but also from the class distribution of predictions by different models in existing SSRE methods. To improve the robustness of the model, we further empower the distillation process with a language model-based AT technique. Extensive experimental results on two public datasets demonstrate that our framework significantly promotes the performance of the base SSRE methods.**

***Index Terms*— Adversarial training (AT), knowledge distillation (KD), relation extraction (RE), semi-supervised learning.**

I. Introduction

RELATION extraction (RE) aims to discover the semantic relations between/among entities from a piece of text, e.g., a sentence or a document. RE tasks can be classified into sentence-level RE [1] and document-level RE [2] types, where two entities belong to one sentence or document, respectively. In this work, we concentrate on the sentence-level RE task. For example, given a sentence "*He threw wood into the bonfire*" and two entities "*e1: wood*" and "*e2: bonfire*," we aim to distinguish the "*entity:destination (e1, e2)*" relation between two entities. RE facilitates transforming massive, unstructured text into structured factual knowledge and has been an active research field due to its broad applications in many machine learning and natural language processing tasks such as knowledge graph construction [3] and biomedical knowledge discovery [4].

The existing RE studies have achieved great success in supervised scenarios by leveraging a huge amount of high-quality labeled data. However, the acquisition of reliable and high-quality labeled data is difficult, expensive, or time-consuming. If only a few labeled data are available, it is a challenge to train a successful RE model. Previous methods for solving the problem of limited labeled data mainly adopt distant supervision (DS) [5] or semi-supervised learning (SSRE) methods [6], [7], [8].

DS has been a hot research topic in recent years. The basic idea of DS is to use a large knowledge base (KB) that stores pairs of entities for various relations to automatically obtain relation labels. It assumes that if two entities belong to a certain relation in KB, all the sentences mentioning these entities indicate the relation. The improvements brought by DS mainly come from expensive KBs. However, it is hard to directly apply the DS RE methods in many applications. This is because DS may bring noisy information, and how to remove the noise is still an unsolved problem, and also because not all the entity pairs can be found in existing KBs.

Another line of research to tackle the problem of insufficient labeled data is semi-supervised relation extraction (SSRE). The SSRE methods follow an essential prerequisite, i.e., "self-training assumption" [9], where the predictions of self-training models, especially those with high confidence, are prone to be correct. Based on this assumption, the existing methods gradually augment the labeled data by iteratively annotating samples with the pseudolabels produced by self-training models. The training does not stop until unlabeled data are exhausted.

The main drawback in early SSRE methods is that the predictions on the unlabeled data might be incorrect, and the problem will be extremely serious when the number of labeled data is limited. To overcome this problem, late SSRE methods often use multiple models [6], [10], [11], which use different seeds for initialization, different parameter spaces for training, or different modules to complement each other. The multiple models collaborate by taking the intersection set of their prediction results to generate high-quality labels. The intuition is that multiple models provide different perspectives on the labeled data, and the labeling bias on augmented data can be partially solved via the collaboration of these models.

Manuscript received 26 April 2022; revised 23 September 2022 and 20 December 2022; accepted 15 March 2023. This work was supported in part by the National Natural Science Foundation of China (NSFC) under Project 62276193; in part by the Key Laboratory of Satellite Information Intelligent Processing and Application Technology under Grant 2022-ZZKY-JJ-16-01; in part by the Joint Laboratory on Credit Science and Technology of China Securities Credit Investment (CSCI), Wuhan University; and in part by the Supercomputing Center of Wuhan University. *(Corresponding author: Tieyun Qian.)*

Wanli Li and Tieyun Qian are with the School of Computer Science, Wuhan University, Wuhan 430072, China (e-mail: wanli.li@whu.edu.cn; qty@whu.edu.cn).

Xuhui Li is with the School of Information Management, Wuhan University, Wuhan 430072, China (e-mail: lixuhui@whu.edu.cn).

Lixin Zou is with the School of Cyber Science and Engineering, Wuhan University, Wuhan 430072, China (e-mail: zoulixin15@gmail.com).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TNNLS.2023.3258967.

Digital Object Identifier 10.1109/TNNLS.2023.3258967

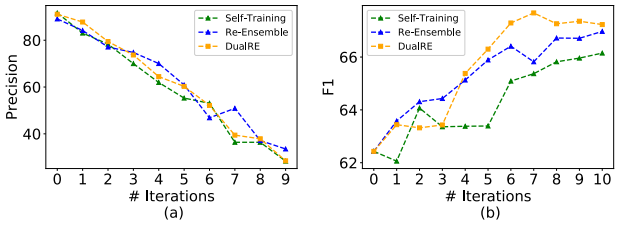

Fig. 1. Quality of augmented data. (a) Precision of the data selected in each iteration. (b) Convergence curve of test F1 score for different methods. The improvement of the models gradually stagnates with the increased number of iterations in training.

Although these late methods have shown improvements over traditional ones, the incorrect pseudolabels on the unlabeled data are inevitable. As shown in Fig. 1, the existing methods using pseudolabels tend to generate low-precision augmented data with the increased number of iterations. While obtaining more reliable labels, current SSRE methods discard the difference set of the prediction results where multiple models assign different labels. This incurs *a big information loss*. On one hand, the class distribution in the difference set of prediction results may still contain correct information. On the other hand, even if none of the results is correct, the predictions convey rich intra- and interclass distribution information.

In this article, we rethink the value of pseudolabels and that of the class distribution and propose a novel framework to address the problem of unreliable pseudolabels on unlabeled data. First, we first introduce the knowledge distillation (KD) technique to exploit the probability distribution over classes in the teacher model to guide the training of the student model. Second, to use the soft label information more accurately, we use a multi-teacher knowledge distillation technique as the backbone of our framework. The ensemble of multiple teachers from different models in SSRE can avoid the bias from a single teacher and help the student model achieve better performance. Third, the teachers' prediction results are used to distinguish the quality of pseudolabels, and the pseudolabels that the teachers agree on are considered high-quality. These pseudolabels will be used for training the student model. Finally, to debias the knowledge learned by the student model and improve the generalization ability of the student, we introduce the adversarial distillation (AD) technique, which uses adversarial examples that are mixed with small perturbations. Such adversarial examples can make the student model adapted to the changes and better absorb the knowledge from teachers. The final proposed framework is called the adversarial knowledge distillation framework.

We conduct extensive experiments on two public datasets. The results demonstrate the superiority of our proposed framework over the state-of-the-art baselines.

In summary, our main contributions are as follows.

1) We frame SSRE within a new paradigm by treating high-quality and low-quality pseudolabels in a separate way.
2) We develop a multi-teacher knowledge distillation framework that unifies the existing SSRE methods and the recently developed KD technique.
3) We empower the multi-teacher distillation (MTD) method with a language model-based AT technique that can further improve the robustness of distillation.

II. Related Work

A. Relation Extraction

Supervised RE is generally categorized into two types: feature-based [12], [13] and kernel-based methods [14]. More recently, pretrained language model (PLM)-based methods become the state-of-the-art [15]. While being powerful, supervised RE models require large amounts of human-annotated data. Some researchers propose DS methods to solve the challenge [16]. Despite their progress, DS methods inevitably suffer from false-positive and false-negative samples. Moreover, it is hard to apply DS for label generation because of the sparse matching results and context-agnostic label noises [7].

Recently, researchers develop semi-supervised RE methods that can alleviate the scarcity problem of labeled data [6]. The motivation behind SSRE is to reduce the manual efforts and to use the information in unlabeled data that are easy to obtain [17], [18]. SSRE combines the advantages of both the supervised and unsupervised paradigms. The main problem in SSRE is the semantic drift [19], i.e., the predictions on the unlabeled data might be incorrect. To address this problem, most SSRE methods adopt an ensemble strategy, where the simplest way is to select samples in the intersection set of multiple classifiers. For example, RE-ensemble [10] selects the samples to expand data based on the agreement of two prediction modules which are independently initialized. DualRE [6] designs a retrieval module to assist the prediction module in generating more accurate annotations. All these methods use pseudolabels to exploit information on unlabeled data.

In this article, we reveal an important low-quality pseudolabel problem of the existing semi-supervised RE methods. The pseudolabels on a portion of training data tend to be inaccurate and the low-quality data will misguide the final model. In other words, the biased label distribution on training data is inevitably generated in the process of semi-supervised learning. Hence, we move one step further and weaken the bias after the last iteration of the semi-supervised process.

B. Knowledge Distillation

KD is originally proposed for model compressing [20], with the basic idea of transferring the knowledge from the large teacher model T to the small student model S. KD has also been successfully applied to various fields, including computer vision [21], natural language understanding such as linguistic acceptability and textual entailment [22], [23] and recommender systems.

KD has not been considered by the existing SSRE methods. We note that a seminal research [24] named KD4NRE distills the knowledge from the soft labels for RE. The differences between our work and [24] are as follows. First, our soft label information comes from the trained semi-supervised model, while part of the soft label information of [24] is derived from statistical information. Second, our method and [24]

present different distillation methods, i.e., MTD and one-teacher distillation, respectively. The method in [24] is suitable for supervised RE tasks. In that scenario, it can use statistics in sufficient training data as soft labels to help distillation and produces a better performance. However, such statistics are unsuitable for semi-supervised relational tasks which have insufficient training data. In this work, we make the first attempt at exploiting KD for SSRE.

C. Adversarial Training

Deep neural networks are highly expressive models that have achieved the state-of-the-art performance in many research fields. However, some neural networks are particularly susceptible to noises and the neural models are pretty sensitive to the input. To avoid the problem, AT [25], [26] was proposed by adding some perturbations to the input while keeping the output unchanged.

AT can alleviate the limitation of KD on the generalization ability of the student model [27]. Several methods have proposed to combine AT with KD to enhance the robustness of the model. However, few of them have effectively explored the impact of adversarial examples since the existing methods usually input the examples into the final model for training. The adversarial samples are automatically generated and not real ones, and thus they may drive the model to learn misleading features. In this article, we develop a new AT method that confines the influence of adversarial examples in the feature encoder rather than directly sending them to the final classifier.

III. Preliminary

A. Problem Definition

Definition 1 (RE): Let $d = [t_1, \ldots, t_m]$ be a sentence with m tokens, and e_1 and e_2 be two entity mentions in d. $R = \{r_1, \ldots, r_{|R|}\}$ is a predefined relation set. The RE[1] task is formulated as a classification problem that determines whether a relation $r \in R$ holds for e_1 and e_2.

Definition 2 (SSRE): Given a set of labeled and unlabeled relation mentions $D_L = \{(d_i, r_i)\}_{i=1}^{|L|}$ and $D_U = \{(d_i)\}_{i=1}^{|U|}$, respectively, the goal of SSRE task is to train a model that fits the labeled data D_L, and it captures the information in the unlabeled data D_U for augmenting the labeled data. The trained model is used to predict the relation of the samples in the unseen test data $D_T = \{(d_i)\}_{i=1}^{|T|}$.

B. Problem Analysis

The existing SSRE methods under "self-training assumption" [9] are inevitably facing the problem of low-quality unlabeled data, i.e., the pseudolabels on those data tend to be wrong. As pointed in [28], labeled samples follow a Borel probability distribution $\Pr(d, r)$, which represents the true class distribution on test data. There is a simplified assumption as follows:

$$\begin{aligned}\Pr(d, r) &= P(r|d)P(d) \\ &= \{P(r|d^L)P(d^L|d) + P(r|d^U)P(d^U|d)\}P(d).\end{aligned} \tag{1}$$

[1]There are three types of RE tasks: sentence-level RE, cross-sentence n-ary RE, and document-level RE. Since prior SSRE methods are conducted on sentence-level tasks, we also adopt this setting.

Since the number of labeled and unlabeled data is fixed, $P(d)$, $P(d^U|d)$, and $P(d^L|d)$ cannot be further optimized. The key to improving SSRE models is the conditional probabilities $p(r|d^L)$ and $p(r|d^U)$. Under the semi-supervised setting, the first probability on labeled data $p(r|d^L)$ is drawn from PR and could be learned by a carefully designed model. For the conditional probability on unlabeled data $p(r|d^U)$, not only the structure of models but also the labeling information matter. Using pseudolabels on low-quality data will lead to inaccurate modeling of conditional probabilities $p(r|d^U)$ because the pseudolabels on low-quality data are likely wrong. Moreover, only using pseudolabels on high-quality data may also limit the model's capability. Thus, a more refined constructing process can improve the existing SSRE methods. Based on this observation, we propose our adversarial MTD (AMTD) framework that treats high-quality and low-quality more delicately.

IV. Methodology

In this section, we present our AMTD framework for SSRE.

A. Model Overview

An overview of our AMTD framework is shown in Fig. 2. It is built upon the existing SSRE methods. We take the models (usually two) in SSRE methods as multiple teachers and then conduct adversarial KD. The framework contains three losses, including a consistency loss of multi-teacher knowledge distillation, a classification loss on the data with high-quality pseudolabels, and a consistency loss on adversarial samples. Specifically, we first use the teachers' prediction results to distinguish high-quality pseudolabels from pseudolabels on unlabeled data, where high-quality pseudolabels are regarded as real labels and low-quality pseudolabels are abandoned. Second, the class distributions on the teachers' predictions for all the data are used for KD. Finally, a new classifier is deployed in the student model for adversarial data to improve the generalization of the student model.

B. Encoder

Before training the models, it is necessary to encode the tokens in input sentences into latent vectors. A variety of encoders have been proposed for RE tasks, such as convolutional neural network (CNN) [29], recurrent neural network (RNN) [13], graph neural network (GNN) [30], and PLM encoders [15], [31]. Among them, PLM encoders can learn universal language knowledge from the large corpus and are beneficial for downstream tasks.

In view of this, we follow the recent practice [15] and use bidirectional encoder representations for transformers (BERT) [32] as PLM for all the SSRE methods. In particular, we insert special tokens "[e1][/e1]" and "[e2][/e2]" at the beginning and the end of the first and second entities, respectively. We also add "[cls]" and "[sep]" tokens at the beginning and the end of the sentence. Given the running example sentence, its input to the encoder is as follows.

"*[cls] The [e1] implant [/e1] is placed into the [e2] jaw bone [/e2] [sep].*"

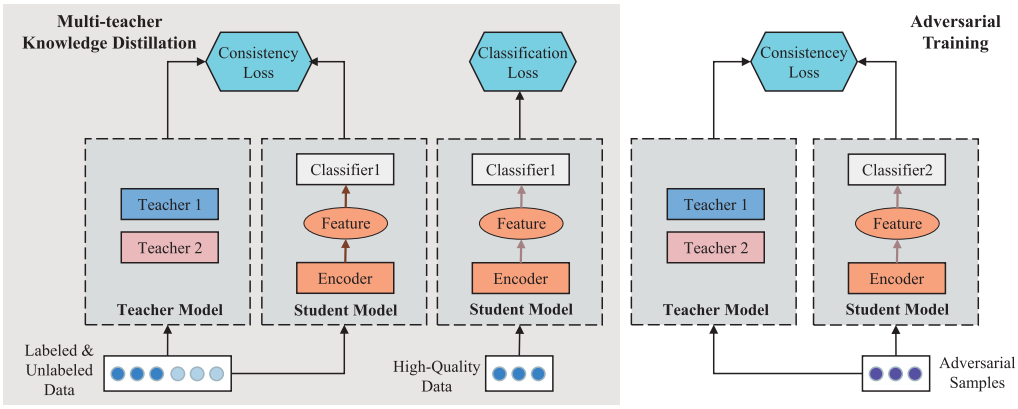

Fig. 2. Overview of our AMTD framework for SSRE. Our AMTD framework consists of two parts: the multi-teacher knowledge distillation and the adversarial training (AT). In MTD, the student model is trained by two losses: a consistency loss with teacher models and a classification loss on high-quality data. In AT, the student model with classifier2 is trained by the consistency loss on adversarial samples.

We feed the PLM-encoded vectors at the corresponding positions of "[e1]" and "[e2]" to the bilinear output layer to obtain the final embedding $\boldsymbol{x}$ of the samples. The process can be defined as follows:

$$\boldsymbol{x} = \text{bilinear}(\boldsymbol{h}_i, \boldsymbol{h}_j) \tag{2}$$

where i and j correspond to the positions of "[e1]" and "[e2], respectively."

C. Multiple Teacher Models

1) Converting Models into Multiple Teachers: In our proposed framework, to be compatible with the existing SSRE methods (termed as *base methods* hereafter), we make the least change to these methods and directly take the multiple models from the last iteration when training the base methods as the multiple teachers. In case the base method like self-training [9] contains one model, we run it two times with different seeds to get two teacher models.

2) Discerning High-/Low-Quality Data: We use the teachers' prediction results to judge the quality of pseudolabels. The pseudolabels with the same annotated labels by different teachers are treated as high-quality, and the data with high-quality pseudolabels or labels are called high-quality data. Formally, we define high-quality data $\mathcal{D}_A$ as follows:

$$\mathcal{D}_A = \mathcal{D}_L \cup \mathcal{D}_I \tag{3}$$

where $\mathcal{D}_L$ is the original labeled data, and $\mathcal{D}_I$ is composed of unlabeled data on which multiple teachers assign the same label. The data except $\mathcal{D}_A$ are regarded as low-quality data $\mathcal{D}_B$. Formally, we define low-quality data $\mathcal{D}_B$ as follows:

$$\mathcal{D}_B = \mathcal{D} - \mathcal{D}_A \tag{4}$$

where $\mathcal{D}$ is all the data including labeled and unlabeled data. As illustrated in Fig. 3, high- and low-quality data are jointly determined by the teachers. If two teachers reach a consensus prediction on a sample, it will be added to the high-quality dataset. Otherwise, it is added to the low-quality dataset.

3) Processing on Different Data: To promote performance, the existing SSRE methods trade off between multiple models and choose the high-confidence label to augment the training data. However, the pseudolabels on high- and low-quality data are dissimilar. Therefore, we propose refined processing on unlabeled data in our AMTD framework by discarding the unreliable pseudolabels on low-quality data.

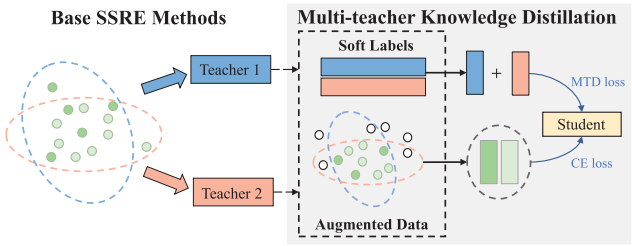

Fig. 3. Illustration of multi-teacher knowledge distillation in our framework. The prediction of teacher models generated from the basic SSRE models is treated as soft labels. MTD loss (the consistency loss) refers to the Kullback–Leibler divergence between the student's predictions and the soft labels on all data, and the CE loss (the classification) is on the high-quality data. Note the green and white dots denote the high- and low-quality samples, respectively.

We construct two losses (classification and consistency loss) on high-quality data and one loss (consistency loss) on low-quality data. Specifically, on high-quality data, since the pseudolabels are reliable and helpful for training, we adopt the classification loss for them. Furthermore, we leverage *dark knowledge* (soft label information) from the predictions on high-quality data. The rationale is that the predictions reflect the judgments of two teachers and contain rich intra- and interclass distribution information. For example, the relationship between "per:city_of_death" and "per:country_of_death" is closer than that between "per:city_of_death" and "org:founded_by." On low-quality data, the pseudolabels might produce misleading knowledge. Such data lie near the decision boundary of models and are hard to be correctly classified. Therefore, we propose only using the soft label information instead of pseudolabels on low-quality data during training.

D. Multi-Teacher Knowledge Distillation

1) Constructing Student Network: The student model S in our AMTD framework has the same neural architecture as the teachers. It consists of an encoder to get the feature of the sentence and a bilinear classifier for prediction. Holding

the same structure ensures the student model can entirely capture the knowledge of teachers.

2) Training Student Model: The constructed student network S is also trained on both $\mathcal{D}_A$ and $\mathcal{D}_B$ with two objectives. One is to minimize the consistency loss between teachers and the student on all data $\mathcal{D}$, i.e., labeled and unlabeled data. The other is to match the ground-truth hard labels on high-quality data $\mathcal{D}_A$, i.e., the classification loss. Formally, we define the loss for training the student network as follows:

$$\mathcal{L}_S = \sum_{i=1}^{|\mathcal{D}_A|} \mathcal{L}_{S,\mathrm{CE}}^i + \lambda \sum_{i=1}^{|\mathcal{D}_A+\mathcal{D}_B|} \mathcal{L}_{\mathrm{MTD}}^i \tag{5}$$

where $\mathcal{L}_{S,\mathrm{CE}}$ denotes the classification loss using one-hot hard labels on high-quality data $\mathcal{D}_A$. Note that the predictions on samples in the high-quality set are also treated as one-hot hard labels. $\mathcal{L}_{\mathrm{MTD}}$ denotes the consistency loss (the distillation loss) using multiple teachers' soft labels, and λ is the hyperparameter to trade off $\mathcal{L}_{\mathrm{CE}}$ and $\mathcal{L}_{\mathrm{MTD}}$.

The classification loss $\mathcal{L}_{\mathrm{CE}}$ is defined as the cross entropy (CE) between the student's predictions and the ground-truth labels

$$\mathcal{L}_{S,\mathrm{CE}} = \sum_{i=1}^{|\mathcal{D}_A|} \mathrm{CE}(G(i), \widetilde{P}_S(i)) \tag{6}$$

where $G(i)$ and $\widetilde{P}_S(i)$ denote the ith element of the ground-truth labels and the student's predictions, respectively.

The consistency loss $\mathcal{L}_{\mathrm{MTD}}$ is defined as the Kullback–Leibler (KL) divergence between the student's predictions $\widetilde{P}_S$ and the soft labels by each of multiple (two) teachers

$$\begin{aligned}\mathcal{L}_{\mathrm{MTD}} &= \sum_{i}^{|D|} \sum_{m\in\{T1,T2\}} KL\big(\widetilde{P}_T^m || \widetilde{P}_S\big) \\ &= \sum_{i}^{|D|} \sum_{m\in\{T1,T2\}} \widetilde{P}_T^m(i) \log\big(\widetilde{P}_T^m(i) / \widetilde{P}_S(i)\big)\end{aligned} \tag{7}$$

where $\widetilde{P}_T^m$ denotes the probability distribution output by the teacher m.

Note that the teacher network generates *a soft class probability* with a converting operation, i.e., raising the temperature of the final softmax layer until the teacher produces a suitably soft set of targets

$$\widetilde{P}_T = \mathrm{softmax}(\widetilde{Z}_T/\tau) \tag{8}$$

where $\widetilde{Z}_T$ is the logits produced by the teacher network. τ is the temperature, and a higher value for τ produces a softer probability distribution over classes.

E. Adversarial Training

In MTD, the teachers' knowledge on the training set (labeled and unlabeled data) is distilled to the students. However, the student may still perform poorly on data that do not appear in the training set because the teachers are trained with limited labeled data in the SSRE scenario. Even small changes in samples (changes that do not affect labels) may mislead the student. To enhance the model robustness, we propose to combine the AT technique with distillation in SSRE, which consists of one adversarial data generation process and one AT process. There are several differences between our approach in this article and adversarially robust distillation (ADR) [27]. First, our motivation is different. Our method aims to leverage the knowledge learned by multiple teachers to enhance the performance of student models, whereas ADR focuses on improving the knowledge and robustness of a single complex teacher network. Second, the tasks and domains we consider are different. Our approach focuses on SSRE, while ADR is primarily applied to image classification.

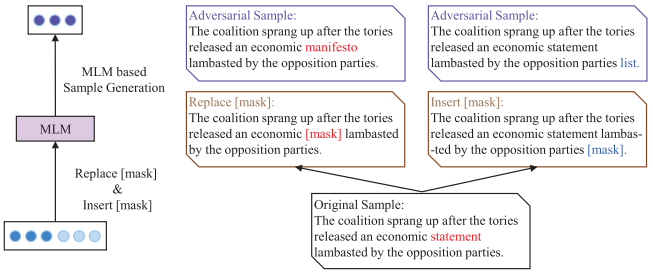

Fig. 4. Illustration of the adversarial sample generation process for one sentence. To generate adversarial samples, the model will replace or insert a special "[mask]" token into the original sample and generates a token for it by the MLM.

1) Adversarial Data Generation: We simply use the pretrained mask language model (MLM) BERT [32] to generate adversarial data. First, we feed the PLM with the original text replaced or inserted by a special token, "[mask]." Then, we fix the rest of the sentence and replace those special mask tokens with the PLM output to construct an AT sample. The ratio of masks we use is consistent with that used by the pretrained language model, i.e., 15%. We present an example of the generation process in Fig. 4.

2) Adversarial Training: We denote the adversarial samples as $\mathcal{D}_C$ which are distinct from the real data and may bring intractable knowledge. For this reason, we develop a specially designed student model which contains two classifiers, classifiers 1 and 2. The training on real and adversarial data goes through classifiers 1 and 2, respectively. By doing this, adversarial data can enhance the model robustness yet does not affect the performance. This is because the student model uses classifier 1 to make the final prediction and the impact of adversarial data does not go beyond the "Encoder" in the AT module in Fig. 2.

We use the MTD loss for training $\mathcal{D}_C$, and the final loss $\mathcal{L}$ of training is as follows:

$$\mathcal{L} = \sum_{i=1}^{|\mathcal{D}_A|} \mathcal{L}_{S,\mathrm{CE}}^i + \lambda \sum_{i=1}^{|\mathcal{D}_A+\mathcal{D}_B|} \mathcal{L}_{\mathrm{MTD}}^i + \gamma \sum_{i=1}^{|\mathcal{D}_c|} \mathcal{L}_{\mathrm{MTD}}^i. \tag{9}$$

V. Experiments

We conduct extensive experiments to verify the effectiveness of our framework. We first introduce the experimental settings and then present the results and analysis.

TABLE I
STATISTICS FOR TWO DATASETS

Dataset	#Train	#Dev	#Test	#Relations	%No_relation
SemEval	7,199	800	2,715	19	17.6
TACRED	68,124	22,631	15,509	42	79.5

A. Dataset

We evaluate our model on two public datasets: SemEval (SemEval-2010 Task 8) [33] and TAC RE dataset (TACRED) [13].

1) **SemEval** is a standard benchmark dataset containing 19 kinds of relations (including no_relation). Note that the relations in SemEval are directed. For example, "entity-destination (e1, e2)" and "entity-destination (e2, e1)" are different relations.
2) **TACRED** is a large-scale RE dataset which includes 41 undirected relation types such as "*per:age*" and an extra "*no_relation*." It is typically used in the annual TAC KB population competition.

The detailed data statistics along with the splits for SemEval and TACRED are shown in Table I. The data split follows the common setting of the existing methods.

B. Compared Methods

To demonstrate the effectiveness of our proposed MTD and AMTD, we compare them with the following supervised, semi-supervised baselines, and the upper bound with golden labels.

The first six methods (1)–(6) are supervised methods. Neural rule grounding (Nero) framework [34] is a rule-based one. These models except NERO use CNN [29], recurrent neural network (RNN) [13], graph convolutional networks (GCNs) [30], KD [24], and BERT [15] as base models, respectively, for supervised RE tasks.

The next five methods (7)–(9) are base SSRE methods. Self-training [9], mean-teacher [11], and RE-ensemble [10] are all general semi-supervised methods. They do not develop any special structure on the SSRE task. DualRE [6] and GradLRE [6] are the state-of-the-art methods on SSRE. DualRE jointly trains a prediction module and a retrieval module to select the top-ranked samples in the intersection set by two modules for data augmentation (DA). GradLRE uses reinforcement learning to judge the correctness of predictions and achieves a good result.

Our proposed MTD and AMTD (12)–(13) are all based on self-training. That is to say, the teacher models in the frameworks are all trained by self-training.

The last one RE-Gold (14) uses the gold labels of the unlabeled data and presents the upper bound for SSRE methods based on BERT_{EM}.

We use the source code provided by the authors of DualRE[2] and GradLRE.[3] Note we use the pointwise variant for DualRE as it performs better than the pairwise one. In addition, we reimplement four base SSRE methods with BERT_{EM} as the encoder to keep pace with the state-of-the-art PLM-based methods.

[2]https://github.com/INK-USC/DualRE
[3]https://github.com/THU-BPM/GradLRE

C. Setup

Following existing SSRE methods [6], [34], we sample 5%, 10%, and 30% training data in SemEval, and 3%, 10%, and 15% training data in TACRED as the labeled set, respectively. About 50% training data in SemEval and TACRED are sampled as unlabeled data whose labels are assumed unavailable for all the models except the BERT with gold labels. For all the compared methods (supervised and semi-supervised), we follow the default hyperparameter settings in the original papers. For all the SSRE methods, we select up to 10% instances of the unlabeled data in the intersection set for DA in each iteration and perform text iterations. The learning rate for the BERT_{EM} encoder and classifier is set to "5e-5" and "1e-4," respectively.

The parameter settings of our model are obtained on the development set. Specifically, the epoch for model training is set to 10, the batch size is 20, and the temperature of distillation is 2.4. We use F_1 as the main metric, and precision and recall as auxiliary metrics to evaluate the performance of all the methods.

D. Main Results

The comparison results on SemEval and TACRED are shown in Tables II and III, respectively.

It can be observed that our AMTD framework with the simplest SSRE method self-training as the base model can achieve significantly better performance than the best baseline on both the datasets and all the ratios of labeled data. This clearly proves the effectiveness and generality of our proposed framework. We see that the trend is more obvious when the ratio is small. This is a very positive finding since we always wish to train a better classifier with less training data. We note that a similar finding also holds for our MTD framework, showing the effects of multi-teacher knowledge distillation. Furthermore, as the size of data increases, the improvement brought by AT gradually decreases. This is reasonable due to the impact of data amount.

Among the first six supervised methods (1)–(6), BERT_{EM} is the best, with remarkable enhancement over other supervised methods. This can be mainly due to the powerful expressive abilities of PLMs. The KD method KD4NRE performs poorly when the proportion of labeled data is small, e.g., 5% and 10% on SemEval. The reason is that KD4NRE is designed for supervised learning and requires statistical information on the full data. Its performance on TACRED with 3% labeled data is not that bad because TACRED is much larger than SemEval. The performance of AGGCN on SemEval is extremely poor as it is based on the undirected connection graph while the relations in SemEval are directed. NERO is stable to different ratios of labeled data since it uses manually specified patterns.

Among the next five semi-supervised methods (7)–(11), the traditional semi-supervised methods (self-training, mean-teacher, and RE-ensemble) perform worse than two recent methods (DualRE and GradLRE) on small-scale datasets, SemEval. This trend is less obvious on TACRED since the large dataset will weaken the performance gap among semi-supervised methods.

TABLE II

COMPARISON RESULTS ON SEMEVAL. THE BEST SCORES OF ALL METHODS ARE IN BOLD, AND THE BEST BASELINE MODELS ARE UNDERLINED. ALL RESULTS ARE THE AVERAGE SCORES OF FIVE RUNS WITH A RANDOM SEED. "‡" INDICATES THE STATISTICALLY SIGNIFICANT IMPROVEMENTS (I.E., TWO-SIDED t-TEST WITH $p < 0.01$) OVER THE BEST BASELINE

	5%			10%			30%		
Methods/LabeledData	Precision	Recall	F_1	Precision	Recall	F_1	Precision	Recall	F_1
(1) PCNN	41.56 ± 2.51	39.30 ± 3.56	40.30 ± 2.49	53.68 ± 1.26	49.87 ± 1.50	51.66 ± 1.38	64.49 ± 0.64	62.81 ± 0.55	63.37 ± 0.42
(2) PRNN	55.65 ± 1.34	53.73 ± 1.25	54.66 ± 0.89	63.47 ± 3.14	61.76 ± 2.20	62.49 ± 0.59	69.66 ± 2.19	68.76 ± 2.60	69.14 ± 1.02
(3) AGGCN	8.26 ± 4.44	5.07 ± 0.49	5.99 ± 1.26	9.70 ± 3.51	4.83 ± 0.30	6.34 ± 0.86	8.69 ± 1.93	5.32 ± 0.25	6.60 ± 0.47
(4) NERO	68.00 ± 1.75	50.63 ± 1.06	58.01 ± 0.19	66.40 ± 0.81	51.96 ± 1.23	58.28 ± 0.52	70.12 ± 0.26	52.28 ± 0.61	59.90 ± 0.27
(5) KD4NRE	55.72 ± 24.80	14.29 ± 1.56	22.45 ± 3.51	72.75 ± 11.43	14.50 ± 0.51	24.12 ± 0.84	64.53 ± 0.91	68.09 ± 0.99	66.26 ± 0.81
(6) BERT_{EM}	68.10 ± 3.08	66.86 ± 4.16	67.43 ± 3.17	76.38 ± 3.82	76.57 ± 5.96	76.45 ± 4.81	84.31 ± 1.91	86.32 ± 0.75	85.29 ± 0.64
(7) Self-Training (ST)	72.75 ± 1.85	74.52 ± 3.18	73.61 ± 2.22	77.44 ± 1.19	83.80 ± 1.82	80.48 ± 0.73	84.06 ± 1.25	86.94 ± 0.74	85.47 ± 0.37
(8) Mean-Teacher	72.37 ± 1.16	74.11 ± 2.11	73.20 ± 0.87	77.97 ± 1.68	84.03 ± 2.21	80.85 ± 0.27	84.89 ± 1.35	86.47 ± 1.47	85.66 ± 0.33
(9) RE-Ensemble	73.45 ± 0.55	75.33 ± 0.94	74.37 ± 0.59	77.86 ± 0.57	84.29 ± 0.92	80.94 ± 0.30	84.66 ± 1.02	86.53 ± 0.68	85.58 ± 0.43
(10) DualRE	73.10 ± 0.81	77.00 ± 1.97	75.15 ± 1.32	80.07 ± 0.84	82.28 ± 2.04	$\underline{81.14 \pm 0.84}$	85.57 ± 1.05	86.39 ± 0.49	$\underline{85.97 \pm 0.56}$
(11) GradLRE	73.21 ± 1.34	80.41 ± 2.38	$\underline{76.64 \pm 1.75}$	76.51 ± 1.29	84.61 ± 1.55	80.34 ± 0.67	82.67 ± 0.38	88.48 ± 0.99	85.47 ± 0.51
(12) ST+MTD	76.46 ± 0.93	79.83 ± 1.80	$78.09 \pm 0.41^{\ddagger}$	80.51 ± 0.50	86.61 ± 0.21	$83.45 \pm 0.32^{\ddagger}$	86.39 ± 0.24	88.84 ± 0.51	$87.60 \pm 0.14^{\ddagger}$
(13) ST+AMTD	78.58 ± 0.95	83.47 ± 0.26	$\mathbf{80.95^{\pm}0.46^{\ddagger}}$	81.30 ± 0.76	87.04 ± 0.98	$\mathbf{84.16^{\pm}0.12^{\ddagger}}$	86.30 ± 0.35	89.03 ± 0.27	$\mathbf{87.65 \pm 0.29^{\ddagger}}$
(14) RE-Gold (BERT_{EM})	87.22 ± 0.42	88.88 ± 0.25	88.04 ± 0.17	86.79 ± 0.29	89.66 ± 0.46	88.20 ± 0.25	87.77 ± 0.50	89.62 ± 0.62	88.68 ± 0.45

TABLE III

COMPARISON RESULTS ON TACRED. THE BEST SCORES OF ALL METHODS ARE IN BOLD, AND THE BEST BASELINE MODELS ARE UNDERLINED. ALL RESULTS ARE THE AVERAGE SCORES OF THREE RUNS WITH A RANDOM SEED. "‡" INDICATES THE STATISTICALLY SIGNIFICANT IMPROVEMENTS (I.E., TWO-SIDED t-TEST WITH $p < 0.01$) OVER THE BEST BASELINE

	3%			10%			15%		
Methods/Labeled Data	Precision	Recall	F_1	Precision	Recall	F_1	Precision	Recall	F_1
(1) PCNN	53.89 ± 3.29	39.93 ± 2.47	45.39 ± 0.78	64.66 ± 3.16	41.84 ± 2.63	50.42 ± 1.00	66.85 ± 0.43	42.90 ± 1.06	52.25 ± 0.67
(2) PRNN	42.85 ± 1.09	39.52 ± 1.65	41.07 ± 0.51	53.44 ± 2.82	51.77 ± 1.88	52.49 ± 0.64	58.88 ± 2.32	51.30 ± 2.04	54.76 ± 0.93
(3) AGGCN	51.95 ± 5.16	38.65 ± 5.51	43.69 ± 1.34	59.80 ± 1.00	51.77 ± 0.95	55.48 ± 0.39	61.30 ± 0.82	52.69 ± 0.30	56.66 ± 0.18
(4) NERO	47.89 ± 0.58	44.21 ± 0.45	45.97 ± 0.41	47.20 ± 0.93	46.94 ± 0.79	47.07 ± 0.61	51.57 ± 1.40	43.98 ± 1.91	47.48 ± 0.92
(5) KD4NRE	54.61 ± 1.78	36.20 ± 0.87	43.52 ± 0.55	58.91 ± 2.35	52.74 ± 2.48	55.59 ± 0.41	56.54 ± 1.52	59.73 ± 1.14	58.07 ± 0.31
(6) BERT_{EM}	59.50 ± 8.22	44.96 ± 3.26	50.86 ± 0.82	61.41 ± 3.74	53.64 ± 3.24	57.13 ± 0.73	63.35 ± 3.11	58.45 ± 3.25	60.69 ± 0.24
(7) Self-Training (ST)	52.34 ± 6.05	53.92 ± 5.73	52.68 ± 0.19	64.23 ± 0.33	54.49 ± 0.39	58.96 ± 0.14	61.90 ± 2.55	61.07 ± 1.79	61.43 ± 0.40
(8) Mean-Teacher	58.14 ± 2.04	47.42 ± 1.43	52.20 ± 0.45	61.35 ± 1.35	57.98 ± 1.32	$\underline{59.60 \pm 0.09}$	62.74 ± 4.59	60.65 ± 4.12	61.47 ± 0.75
(9) RE-Ensemble	53.77 ± 3.11	51.18 ± 2.03	52.37 ± 1.06	61.15 ± 0.97	57.25 ± 0.41	59.13 ± 0.40	62.27 ± 0.94	61.03 ± 0.97	$\underline{61.63 \pm 0.34}$
(10) DualRE	57.66 ± 1.20	50.02 ± 0.81	$\underline{53.56 \pm 0.52}$	64.02 ± 2.53	55.64 ± 1.49	59.50 ± 0.79	62.79 ± 1.48	60.38 ± 0.43	61.55 ± 0.61
(11) GradLRE	55.84 ± 14.53	39.51 ± 7.10	44.77 ± 1.43	53.72 ± 0.99	62.66 ± 0.16	57.84 ± 0.52	56.77 ± 4.39	61.97 ± 2.03	59.14 ± 1.50
(12) ST+MTD	63.55 ± 2.26	49.17 ± 1.67	$55.40 \pm 0.24^{\ddagger}$	70.45 ± 1.43	55.37 ± 0.62	$62.00 \pm 0.43^{\ddagger}$	65.83 ± 1.91	62.83 ± 1.23	$64.27 \pm 0.42^{\ddagger}$
(13) ST+AMTD	61.94 ± 0.36	54.25 ± 0.68	$\mathbf{57.84 \pm 0.54^{\ddagger}}$	63.47 ± 0.89	62.20 ± 1.91	$\mathbf{62.81 \pm 0.64^{\ddagger}}$	65.85 ± 0.35	63.51 ± 0.19	$\mathbf{64.66 \pm 0.27^{\ddagger}}$
(14) RE-Gold (BERT_{EM})	68.33 ± 1.02	61.78 ± 1.57	64.88 ± 0.84	68.83 ± 1.87	61.60 ± 1.92	64.98 ± 0.37	68.34 ± 2.03	62.28 ± 2.13	65.13 ± 0.40

TABLE IV

COMPARISON RESULTS OF DIFFERENT BASE SSRE METHODS ON SEMEVAL AND TACRED. THE RESULTS ON SEMEVAL AND TACRED ARE AVERAGED OVER FIVE AND THREE RUNS WITH RANDOM SEEDS, RESPECTIVELY. "‡" DENOTES THE STATISTICALLY SIGNIFICANT IMPROVEMENTS (I.E., TWO-SIDED t-TEST WITH $p < 0.01$) OVER THE CORRESPONDING BASE SSRE METHOD

	SemEval			TACRED		
Methods/LabeledData	5% (F_1)	10% (F_1)	30% (F_1)	3% (F_1)	10% (F_1)	15% (F_1)
(1) Self-Training	$73.61_{\pm 2.22}$	$80.48_{\pm 0.73}$	$85.47_{\pm 0.37}$	$52.68_{\pm 0.19}$	$58.96_{\pm 0.14}$	$61.43_{\pm 0.40}$
+ Our MTD	$78.09^{\ddagger}_{\pm 0.41}$	$83.45^{\ddagger}_{\pm 0.32}$	$87.60^{\ddagger}_{\pm 0.14}$	$55.40^{\ddagger}_{\pm 0.24}$	$62.00^{\ddagger}_{\pm 0.43}$	$64.27^{\ddagger}_{\pm 0.42}$
(2) RE-Ensemble	$74.37_{\pm 0.59}$	$80.94_{\pm 0.30}$	$85.58_{\pm 0.43}$	$52.37_{\pm 1.06}$	$59.13_{\pm 0.40}$	$61.63_{\pm 0.34}$
+ Our MTD	$78.04^{\ddagger}_{\pm 0.51}$	$83.08^{\ddagger}_{\pm 0.29}$	$87.47^{\ddagger}_{\pm 0.19}$	$56.18^{\ddagger}_{\pm 0.41}$	$62.20^{\ddagger}_{\pm 0.20}$	$64.34^{\ddagger}_{\pm 0.61}$
(3) DualRE	$75.15_{\pm 1.32}$	$81.14_{\pm 0.84}$	$85.97_{\pm 0.56}$	$53.56_{\pm 0.52}$	$59.50_{\pm 0.79}$	$61.55_{\pm 0.61}$
+ Our MTD	$78.08^{\ddagger}_{\pm 0.64}$	$84.06^{\ddagger}_{\pm 0.14}$	$87.62^{\ddagger}_{\pm 0.31}$	$56.45^{\ddagger}_{\pm 0.12}$	$62.17^{\ddagger}_{\pm 0.30}$	$64.29^{\ddagger}_{\pm 0.55}$
(4) BERT w. gold labels	$88.04^{\ddagger}_{\pm 0.17}$	$88.20^{\ddagger}_{\pm 0.25}$	$88.68^{\ddagger}_{\pm 0.45}$	$64.88_{\pm 0.84}$	$64.98_{\pm 0.37}$	$65.13_{\pm 0.40}$

We also find that the KD-based method KD4NRE does not perform well on semi-supervised tasks. Indeed, KD4NRE is the state-of-the-art method in the supervised RE task. It first uses large-scale data to train a teacher model, and then the teacher cooperates with the statistical information on data to jointly teach a student model. However, in the semi-supervised RE task, the statistics information on small-scale labeled data becomes inaccurate, which causes the KD4NRE model to fail.

Our framework is able to be applied to various base SSRE methods. We show the performance of the framework based on different SSRE methods in Table IV.

It can be observed that our MTD framework can significantly enhance the performance of the base SSRE methods on both the datasets and all the ratios of labeled data. This clearly proves the effectiveness and generality of our proposed framework. Moreover, we see that the trend is more obvious when the ratio is small. This is a very positive finding since we always wish to train a better classifier with less training data.

VI. DEEP ANALYSIS

To get a deep insight into our proposed AMTD framework, we conduct a series of experiments, including the ablation study, hyperparameter and loss landscape analysis, case study, and complexity analysis.

TABLE V

TEACHER AND STUDENT STRUCTURE ANALYSIS ON SEMEVAL. THE BEST SCORES OF ALL METHODS ARE IN BOLD. ALL RESULTS ARE THE AVERAGE SCORES OF FIVE RUNS WITH A RANDOM SEED

	Teacher: Bert-base			Teacher: Bert-large		
Methods/LabeledData	5%	10%	30%	5%	10%	30%
(1) Teacher (ST)	73.61 ± 2.22	80.48 ± 0.73	85.47 ± 0.37	77.86 ± 1.99	82.02 ± 1.84	86.28 ± 0.44
(2) ST+AMTD (bert-base)	80.95 ± 0.46	84.16 ± 0.12	87.65 ± 0.29	82.23 ± 0.41	85.74 ± 0.48	$\mathbf{87.72 \pm 0.34}$
(3) ST+AMTD (bert-large)	80.06 ± 0.79	84.48 ± 0.36	87.45 ± 0.15	$\mathbf{82.29 \pm 0.58}$	$\mathbf{85.96 \pm 0.58}$	87.67 ± 0.36

TABLE VI

ABLATION STUDY ON SEMEVAL AND TACRED. ↓ DENOTES A DROP OF F1 SCORE

Methods/LabeledData	10%SemEval F1	10%TACRED F1
AMTD	$84.16_{\pm 0.12}$	$62.81_{\pm 0.64}$
MTD	$83.45_{\pm 0.32}$ ↓	$62.00_{\pm 0.43}$ ↓
MTD_B	$83.01_{\pm 0.40}$ ↓	$61.47_{\pm 0.75}$ ↓
Intersection	$82.89_{\pm 0.41}$ ↓	$60.98_{\pm 0.23}$ ↓
Distillation_O	$82.01_{\pm 0.69}$ ↓	$61.79_{\pm 0.25}$ ↓
Self-Training	$80.48_{\pm 0.73}$ ↓	$58.96_{\pm 0.14}$ ↓

A. Ablation Study

We design two ablation studies, with 10% labeled data on SemEval and TACRED, to examine the impacts of different components. All the ablation studies are based on the simple self-training method.

1) **MTD** uses classification loss and consistency loss on high-quality data, i.e., we remove the AT process.
2) **MTD_B** uses cross-entropy loss on high-quality data and MTD loss on low-quality data under the MTD framework. In this case, only the knowledge on low-quality data is modeled. Through this experiment, we wish to see whether soft labels on high-quality data contain useful information.
3) **Intersection** trains a separate model based on pseudolabels predicted by two teacher models on the high-quality data $\mathcal{D}_A$ (the intersection set) without the MTD loss.
4) **Distilation_O** uses one teacher for distillation and trains a model based on pseudolabels predicted by the teacher model on all the labeled and unlabeled data, i.e., we keep the partial distillation by treating one teacher's predictions as soft labels to demonstrate the effect of multi-teacher techniques.
5) **Self-Training** is the original SSRE method which contains only one model.

The results for ablation studies on SemEval and TACRED are shown in Table VI. We make the following notes for these results.

First, all the variants with reduced components cause performance drops. This demonstrates that all the components contribute to the entire framework. For example, the better performance of "AMTD" than "MTD" proves the impact of AT.

Second, by comparing results for "AMTD" and "MTD" with those for other methods, the effects of distillation can be confirmed by the superior performance of these two models which both contain the complete distillation component. Moreover, "MTD" outperforms "MTD_B" because soft label information on high-quality data contains interclass information that can help improve models.

TABLE VII

IMPACTS OF THE NUMBER OF TEACHERS ON 10% SEMEVAL. THE BEST SCORES OF ALL METHODS ARE IN BOLD

Number of teachers	P	R	F1
1 Teacher	$80.23_{\pm 0.93}$	$83.90_{\pm 1.86}$	$82.01_{\pm 0.69}$
2 Teacher	$80.51_{\pm 0.50}$	$86.61_{\pm 0.21}$	$83.45_{\pm 0.32}$
3 Teacher	$80.51_{\pm 0.44}$	$87.74_{\pm 0.34}$	$\mathbf{83.97_{\pm 0.28}}$
5 Teacher	$80.13_{\pm 0.31}$	$86.50_{\pm 0.43}$	$83.19_{\pm 0.13}$

Third, "Intersection" produces a good performance, showing that the models from two SSRE modules can provide high-quality labeling information. "Distilation_O" is worse than the complete MTD framework, showing that the knowledge distilled from one teacher is less effective than that from two teachers.

We also see that "Distilation_O" is a bit inferior to "Intersection" on SemEval but it is better on TACRED. This can be due to the property of the two datasets. Taking a close look at the data, we find that the proportion of intersection data to unlabeled data is 83.56% on SemEval and 91.22% on TACRED, while the ratio of negative (no_relation) samples is 14.65% in SemEval and 82.89% in TACRED. This suggests that the intersection set in SemEval contains more high-quality samples.

B. Analysis on Teachers and Students Structure

We design experiments to discuss the effects of the teachers' configuration on students, including the impact of different teacher network structures, and the number of teachers.

First, to examine the influence of network structure on the model, we increase the number of layers, attention heads, and dimensions of the encoder. The basic encoder is changed from the "Bert-base" to the "Bert-large" model. The results are shown in Table V. We see that a more complex model structure can improve the performance of the teachers. We can also find the influence of better teachers on students. There is no doubt that good teachers can also improve the performance of student models. Interestingly, more complex student structures do not improve the student's performance. It infers that the current structure of "Bert-base" is sufficient to learn the knowledge of teachers, and a more complex structure will not bring further improvement under the same teachers.

Second, we also examine the effect of the number of teacher models on student models in Table VII. A too large teacher number (e.g., 5) may degrade the performance due to the decreased number of samples in the intersection set and it also increases the complexity.

C. Analysis on AD

To better understand the advantages of AD, we compare our proposed AD with the traditional DA. In this experiment,

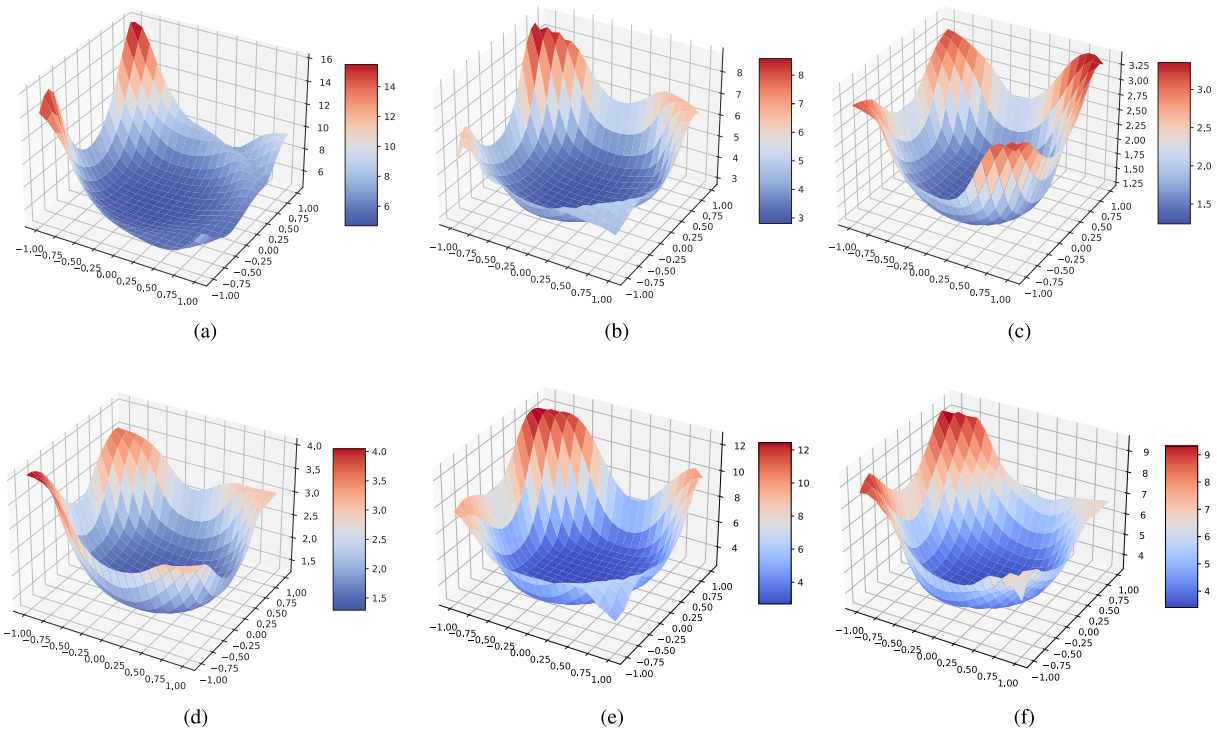

Fig. 5. Visualization of the loss landscape. (a) Self-training. (b) Intersection_S. (c) Distillation_O. (d) Self-training + MTD. (e) Self-training + AT. (f) Self-training + AMTD.

TABLE VIII
COMPARISON RESULTS OF AD AND DA ON 10% SEMEVAL. THE BEST SCORES OF ALL METHODS ARE IN BOLD

Methods	P	R	F1
ST+MTD	$80.51_{\pm 0.50}$	$86.61_{\pm 0.21}$	$83.45_{\pm 0.32}$
ST+MTD+DA	$79.61_{\pm 0.46}$	$88.43_{\pm 0.37}$	$83.79_{\pm 0.37}$
ST+AMTD	$81.30_{\pm 0.76}$	$87.04_{\pm 0.98}$	$\mathbf{84.16_{\pm 0.12}}$

AD and DA use soft labels and hard labels, respectively, to help train the student model. As shown in Table VIII, the performance of DA is worse than that of AD, which means that the DA can improve the generalization performance of models without introducing new knowledge, while AD methods can improve more with knowledge from teachers.

D. Analysis on Loss Landscape

A recent study [35] shows that the loss landscape can support the analysis of KD methods. We use the state-of-the-art landscape visualization technique [36] to plot the loss surface of four representative methods and their variants. The results on SemEval are shown in Fig. 5.

It is clear that our self-training + MTD produces the most flatter surface around the local minima among the first four methods [Fig. 5(a)–(d)]. Distillation_O has similar results. Meanwhile, the surfaces for two methods without distillation (self-training and intersection_S) are much sharper. These results show that the distillation can produce a better loss landscape. This may be caused by the rich interclass relationships contained in KD.

For two methods using AT [Fig. 5(e) and (f)], the height of their loss landscape becomes a bit larger due to AT. However, their curvature remains to be smooth, showing better generalization ability.

TABLE IX
TEMPERATURE FACTOR ANALYSIS BASED ON MULTI-TEACHER. THE BEST SCORES OF ALL METHODS ARE IN BOLD

T	1.5	2.0	2.5	3.0	3.5	Optimal
Teacher 1	79.04	78.18	78.44	78.64	78.51	–
Teacher 2	78.66	78.51	78.72	78.89	79.07	–
Multi-teacher	79.36	**79.79**	79.66	79.69	79.76	79.32

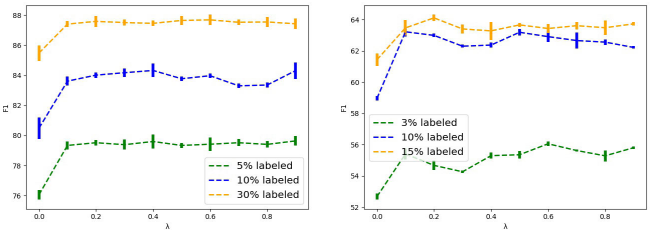

Fig. 6. Impacts of the hyperparameter λ. (a) SemEval. (b) TACRED.

It can be found that the optimal setting for λ is about 0.3 on SemEval and 0.5 on TACRED. When λ is set to 0, the distillation loss for soft labels does not work. In such a case, the framework degenerates into a simple model using only the ground-truth hard labels for training. When λ is set to 1, the framework is equivalent to a variant using only the soft labels for training. Both these can hurt the performance of the model.

E. Analysis on Hyper-Parameter

In the framework, the temperature τ and λ are critical. τ can affect the distribution of soft labels and will further affect

TABLE X

CASE STUDY. THE RED AND BLUE TOKENS DENOTE THE SUBJECT (e_1) AND OBJECT (e_2) ENTITY, RESPECTIVELY. THE TOP THREE PREDICTIONS ARE PRESENTED IN AN ASCENDING ORDER FOR TEACHER AND STUDENT MODELS, AND THE GROUND-TRUTH LABELS ARE UNDERLINED

Instance (I)	**Teacher1 (T1)**		**Teacher2 (T2)**		**Student (S)**	
I1: They include sending e-mails to remind customers about abandoned items.	Content-Container(e2,e1)	0.109	Instrument-Agency(e2,e1)	0.003	Entity-Origin(e1,e2)	0.026
	Message-Topic(e1,e2)	0.112	Entity-Origin(e1,e2)	0.004	Entity-Destination(e1,e2)	0.051
	Entity-Origin(e1,e2)	0.232	Message-Topic(e1,e2)	0.987	Message-Topic(e1,e2)	0.786
I2: Tree roots that grow on the surface are difficult to mow or walk over and can effect the growth and health of nearby grass and groundcovers.	Member-Collection(e2,e1)	0.106	Member-Collection(e1,e2)	0.170	Cause-Effect(e1,e2)	0.032
	Cause-Effect(e1,e2)	0.158	Component-Whole(e2,e1)	0.233	Entity-Origin(e2,e1)	0.113
	Component-Whole(e2,e1)	0.308	Component-Whole(e1,e2)	0.374	Component-Whole(e2,e1)	0.644
I3: " Partnering to Achieve Greater Effectiveness in Preventing Blindness," Kathy Spahn, President and CEO, Helen Keller International.	org:top_members/employees	0.055	org:members	0.006	per:title	
	no_relation	0.179	no_relation	0.448	org:top_members/employees	0.029
	per:title	0.742	org:founded_by	0.530	no_relation	0.156
						0.643

the model performance. λ is used for balancing the loss of hard labels and that of soft labels.

To analyze temperature as comprehensively as possible, we first remove the influence of soft labels on multiple teachers and only use the soft labels of a single teacher. In this way, we can find the optimal temperature in the case of a single teacher, and then assign the optimal temperature to each teacher, and further observe whether setting the optimal temperature for each teacher will bring improvement to the MTD. As shown in Table IX, temperature factors can affect student performance, but setting the optimal temperature for each teacher will not improve student performance in AMTD.

We show the impacts of the hyperparameter λ on SemEval and TACRED with 10% labeled data in Fig. 6. To highlight the impacts of distillation parameters, we remove the effect of AT in our framework.

F. Case Study

We perform a case study by presenting the soft label output from the teacher models and the prediction results of the student model on several test samples from SemEval and TACRED. The results are shown in Table X.

The first two instances I1 and I2 in Table X are from SemEval. In I1, the teacher T1 assigns the wrong label "Entity-Origin(e1,e2)" with a low confidence. Meanwhile, T2 assigns the correct label "Message-Topic(e1,e2)" with a very high confidence. In I2, both T1 and T2 successfully identify the "Component-Whole" relation. However, T2 assigns a wrong direction and the correct prediction ranks second. In both the instances, the student S can make the correct prediction, indicating the effects of distillation from two teachers. If there is only one teacher, it might be hard for S to acquire sufficient knowledge. If only T1 or T2 is chosen as the teacher model, it might be hard for S to acquire the correct label.

The last instance I3 in Table X is from TACRED. In I3, both teachers T1 and T2 make wrong yet different predictions. However, their second highest scores are the same and correct. The trained student network S inherits the knowledge distilled from teachers and then surpasses the teachers.

In summary, the student model in our proposed framework learns both the common and different knowledge from the two teacher models. From the view of framework structure, the reason for the correct judgment predicted by the student model can be due to multiple teachers' collaboration.

TABLE XI

COMPLEXITY ANALYSIS. h = hour, $M = 1 \times 10^6$

	10%SemEval		10%TACRED	
	time	space	time	space
Self-Training	0.41h	124M	3.74h	124M
Mean-Teacher	0.98h	249M	10.15h	251M
RE-Ensemble	0.97h	249M	10.47h	250M
DualRE	0.98h	249M	10.66h	250M
ST+MTD	0.91h	124M	8.24h	124M
ST+AMTD	1.08h	124M	9.80h	124M

G. Analysis on Computational Cost

To prove that the improvement of our model does not incur big computational cost, we perform a complexity analysis. The results on a 24-GB NVIDIA RTX 3090 GPU are shown in Table XI.

From Table XI, we can see that self-training has the smallest time cost because it only trains one model for semi-supervised learning. Our ST + MTD is the most efficient among all other methods. Recall that all these SSRE methods except self-training have two models, and the main time cost is for the iterative semi-supervised learning while MTD only needs to train the student model. In addition, we find that AMTD has the biggest time cost on SemEval due to its AT procedure. However, on a large dataset like TACRED, it is still in the middle since other methods require more time to train their model. Furthermore, mean-teacher and RE-ensemble consist of two models and adopt various strategies to select high-confidence augmented data, which leads to their running time being doubled compared with the self-training method. DualRE uses various training losses and special structures, and thus it has a bigger time cost than other models.

As for the running space, we find that the integration of our MTD and AMTD does not increase the space cost upon the base self-training method, and it is much small than that of other SSRE methods. This can be due to the sequential training of our framework which only contains one model at a time during training. Concretely, in our proposed framework, we first run the self-training method twice to obtain two

teacher models and then save the teacher's output. During this process, the space is the same as that of self-training. The student is directly trained by the saved teacher output without loading the teacher model. Therefore, the student training does not require extra space costs.

VII. Conclusion

In this article, we propose a novel framework for SSRE. The key observation is that the existing SSRE methods neglect the class distribution information hidden in the multiple models' predictions. Based on this observation, we first design an MTD framework to transfer the distribution knowledge from two teacher networks to the student network. MTD is simple and general, and it can be easily integrated with the existing SSRE methods. We also develop an AT technique to further improve the robustness of the distillation process. Extensive experiments on two popular datasets verify that our framework can significantly improve the performance of the base SSRE methods.

Acknowledgment

The numerical calculations in this article have been done on the supercomputing system in the Supercomputing Center of Wuhan University.

References

[1] I. Hendrickx et al., "SemEval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals," in *Proc. ACL*, 2010, pp. 33–38.

[2] Y. Yao et al., "DocRED: A large-scale document-level relation extraction dataset," in *Proc. ACL*, 2019, pp. 764–777.

[3] C. D. Sa et al., "Incremental knowledge base construction using deepdive," *VLDB J.*, vol. 26, no. 1, pp. 81–105, 2017.

[4] C. Quirk and H. Poon, "Distant supervision for relation extraction beyond the sentence boundary," in *Proc. 15th Conf. Eur. Chapter Assoc. Comput. Linguistics (EACL)*, 2017, pp. 1171–1182, doi: 10.18653/v1/e17-1110.

[5] M. Reiplinger, M. Wiegand, and D. Klakow, "Relation extraction for the food domain without labeled training data—Is distant supervision the best solution?" in *Proc. 9th Int. Conf. Natural Lang. Process. (NLP PolTAL)* (Lecture Notes in Computer Science), vol. 8686. Switzerland: Springer, 2014, pp. 345–357.

[6] H. Lin, J. Yan, M. Qu, and X. Ren, "Learning dual retrieval module for semi-supervised relation extraction," in *Proc. World Wide Web Conf.*, May 2019, pp. 1073–1083.

[7] X. Hu, C. Zhang, F. Ma, C. Liu, L. Wen, and P. S. Yu, "Semi-supervised relation extraction via incremental meta self-training," in *Proc. Findings Assoc. Comput. Linguistics (EMNLP)*, 2021, pp. 487–496.

[8] Z. Chen and T. Qian, "Enhancing aspect term extraction with soft prototypes," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2020, pp. 2107–2117.

[9] C. Rosenberg, M. Hebert, and H. Schneiderman, "Semi-supervised self-training of object detection models," in *Proc. Seventh IEEE Workshops Appl. Comput. Vis.*, Jan. 2005, pp. 29–36.

[10] G. French, M. Mackiewicz, and M. H. Fisher, "Self-ensembling for visual domain adaptation," in *Proc. 6th Int. Conf. Learn. Represent. (ICLR)*, 2018, pp. 1–20.

[11] A. Tarvainen and H. Valpola, "Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results," in *Proc. Adv. Neural Inf. Process. Syst., Annu. Conf. Neural Inf. Process. Syst. (NeurIPS)*, 2017, pp. 1195–1204.

[12] Y. S. Chan and D. Roth, "Exploiting syntactico-semantic structures for relation extraction," in *Proc. Annu. Meeting Assoc. Comput. Linguistics (ACL)*, 2011, pp. 551–560.

[13] Y. Zhang, V. Zhong, D. Chen, G. Angeli, and C. D. Manning, "Position-aware attention and supervised data improve slot filling," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2017, pp. 35–45.

[14] T. T. Nguyen, A. Moschitti, and G. Riccardi, "Convolution kernels on constituent, dependency and sequential structures for relation extraction," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2009, pp. 1378–1387.

[15] L. B. Soares, N. FitzGerald, J. Ling, and T. Kwiatkowski, "Matching the blanks: Distributional similarity for relation learning," in *Proc. Annu. Meeting Assoc. Comput. Linguistics (ACL)*, 2019, pp. 2895–2905.

[16] M. Mintz, S. Bills, R. Snow, and D. Jurafsky, "Distant supervision for relation extraction without labeled data," in *Proc. Annu. Meeting Assoc. Comput. Linguistics (ACL)*, 2009, pp. 1003–1011.

[17] R. Gabbard, M. Freedman, and R. M. Weischedel, "Coreference for learning to extract relations: Yes Virginia, coreference matters," in *Proc. Annu. Meeting Assoc. Comput. Linguistics (ACL)*, 2011, pp. 288–293.

[18] H.-T. Zhang, M.-L. Huang, and X.-Y. Zhu, "A unified active learning framework for biomedical relation extraction," *J. Comput. Sci. Technol.*, vol. 27, no. 6, pp. 1302–1313, Nov. 2012.

[19] J. R. Curran, T. Murphy, and B. Scholz, "Minimising semantic drift with mutual exclusion bootstrapping," in *Proc. PACLING*, 2007, pp. 172–180.

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

[21] Y. Zhang, T. Xiang, T. M. Hospedales, and H. Lu, "Deep mutual learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4320–4328.

[22] K. Clark, M.-T. Luong, U. Khandelwal, C. D. Manning, and Q. V. Le, "BAM! Born-again multi-task networks for natural language understanding," in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*, 2019, pp. 5931–5937.

[23] H. Fu et al., "LRC-BERT: Latent-representation contrastive knowledge distillation for natural language understanding," in *Proc. 35th AAAI Conf. Artif. Intell. (AAAI)*, 2021, pp. 1–9.

[24] Z. Zhang et al., "Distilling knowledge from well-informed soft labels for neural relation extraction," in *Proc. 34th AAAI Conf. Artif. Intell. (AAAI)*, 2020, pp. 9620–9627.

[25] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2015, pp. 1–11.

[26] C. Szegedy et al., "Intriguing properties of neural networks," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, Y. Bengio and Y. LeCun, Eds. Ithaca, NY, USA: arXiv.org, 2014, pp. 1–10.

[27] M. Goldblum, L. Fowl, S. Feizi, and T. Goldstein, "Adversarially robust distillation," in *Proc. 34th AAAI Conf. Artif. Intell. (AAAI)*, 2020, pp. 3996–4003.

[28] J. Huang, A. J. Smola, A. Gretton, K. M. Borgwardt, and B. Schölkopf, "Correcting sample selection bias by unlabeled data," in *Proc. Adv. Neural Inf. Process. Syst., 20th Annu. Conf. Neural Inf. Process. Syst. (NeurIPS)*, 2006, pp. 601–608.

[29] D. Zeng, K. Liu, Y. Chen, and J. Zhao, "Distant supervision for relation extraction via piecewise convolutional neural networks," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2015, pp. 1753–1762.

[30] Z. Guo, Y. Zhang, and W. Lu, "Attention guided graph convolutional networks for relation extraction," in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*, 2019, pp. 241–251.

[31] I. Yamada, A. Asai, H. Shindo, H. Takeda, and Y. Matsumoto, "LUKE: Deep contextualized entity representations with entity-aware self-attention," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2020, pp. 1–13.

[32] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics: Hum. Lang. Technol. (NAACL-HLT)*, 2019, pp. 4171–4186.

[33] I. Hendrickx et al., "SemEval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals," 2019, *arXiv:1911.10422*.

[34] W. Zhou et al., "NERO: A neural rule grounding framework for label-efficient relation extraction," in *Proc. Web Conf. (WWW)*, 2020, pp. 2166–2176.

[35] S. Mirzadeh, M. Farajtabar, A. Li, N. Levine, A. Matsukawa, and H. Ghasemzadeh, "Improved knowledge distillation via teacher assistant," in *Proc. 34th AAAI Conf. Artif. Intell. (AAAI)*, 2020, pp. 5191–5198.

[36] H. Li, Z. Xu, G. Taylor, C. Studer, and T. Goldstein, "Visualizing the loss landscape of neural nets," in *Proc. Adv. Neural Inf. Process. Syst., Annu. Conf. Neural Inf. Process. Syst. (NeurIPS)*, 2018, pp. 6391–6401.