Self-distilled Transitive Instance Weighting for Denoised Distantly Supervised Relation Extraction

Anonymous ACL submission

Abstract

The widespread existence of wrongly labeled instances is a challenge to distantly supervised relation extraction. Most of the previous works are trained in a bag-level setting to alleviate such noise. However, sentence-level training better utilizes the information than bag-level training, as long as combined with effective noise alleviation. In this work, we propose a novel Transitive Instance Weighting mechanism integrated with the self-distilled BERT backbone, utilizing information in the intermediate outputs to generate dynamic instance weights for denoised sentence-level training. By down-weighting wrongly labeled instances and discounting the weights of easy-to-fit ones, our method can effectively tackle wrongly labeled instances and prevent overfitting. Experiments on both held-out and manual datasets indicate that our method achieves state-of-the-art performance and consistent improvements over the baselines.

1 Introduction

Distantly Supervised Relation Extraction (DSRE) (Mintz et al., 2009) is designed to automatically annotate the sentences mentioning the entity pairs, which enables a significant way of constructing large-scale datasets. However, distant supervision (DS) works under an unrealistic assumption that all sentences mentioning the same entity pair express the same relation. This introduces many noisy (wrongly labeled) instances into the dataset. To tackle this challenge, previous works mostly adopt the bag-level setting as shown at the top of Figure 1, where the vector representations of sentences are aggregated as the bag-level representation using multi-instance learning (MIL) (Riedel et al., 2010), and the prediction is thus produced from the bag representation. The optimization is conducted at the bag level to minimize the loss of bag prediction. Only a small subset of previous works leverage the sentence-level setting (Zhang et al., 2019b; Liu et al., 2020a) as in the bottom of Figure 1, where the sentence-level predictions are produced and then aggregated into the bag prediction. In fact, sentence-level training can directly optimize the loss from each sentence, enabling higher information utilization than bag-level training. However, sentence-level training is vulnerable to the noise brought by DS, which limits its application. Therefore, sentence-level training should be combined with effective noise-alleviation mechanisms to improve its robustness.

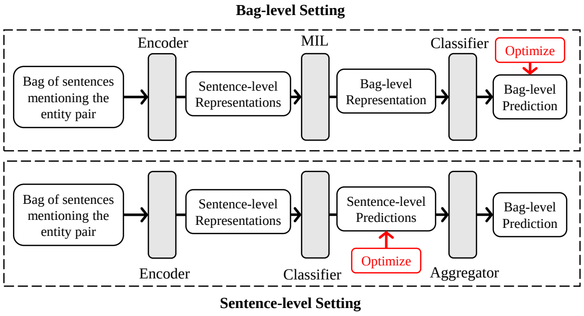
Bag-level Setting
Encoder
MIL
Classifier
Optimize
Bag of sentences mentioning the entity pair
Sentence-level Representations
Bag-level Representation
Bag-level Prediction
Bag of sentences mentioning the entity pair
Sentence-level Representations
Sentence-level Predictions
Optimize
Bag-level Prediction
Encoder
Classifier
Aggregator
Sentence-level Setting

Figure 1: The bag-level and sentence-level pipelines of DSRE.

The mainstream encoders of DSRE models are Piecewise Convolutional Neural Network (PCNN) (Zeng et al., 2015) and Recurrent Neural Network (RNN) (Zhou et al., 2016; Liu et al., 2018) over the years. It is reasonable for most previous works to take the simple encoder as a black box and only utilize its final output during training and inference. However, as large models like BERT (Devlin et al., 2019) become popular in recent years, the information within the outputs from their intermediate layers is a non-trivial source of knowledge but is rarely discussed in DSRE. In this work, we apply self-distillation to extract intermediate information as output probabilities and utilize them to denoise from wrong labels. Furthermore, we use soft target selection and set up transitive knowledge passing among the students to alleviate the effects of noisy target probabilities from the teacher.

The instances in DSRE can be roughly divided into easy, hard and noisy ones. Both easy and hard instances are correctly labeled but the model learns from hard instances slower (Huang et al., 2021). Noisy instances have wrong labels and can be further divided into False Positives (FPs) and False Negatives (FNs). FPs are instances with NA relation but are wrongly labeled as non-NA relations by DS, while FNs are non-NA instances wrongly labeled as NA. We hope to avoid learning from noisy instances since they contain misleading information. Moreover, we also need to avoid overfitting to easy instances to improve the learning of deeper knowledge.

To tackle the above challenges, we propose a novel **Transitive Instance Weighting (TIW)** mechanism for denoised sentence-level training in DSRE. Firstly, we apply self-distillation to directly reuse the knowledge of the teacher model for further denoising in students and set up a transitive way to share knowledge among students. Secondly, we leverage the TIW mechanism to generate robust instance weights to reduce noise and overfitting during distillation. TIW considers two factors in the generation of instance weights: **Uncertainty** (Liu et al., 2020b) for overfitting prevention and general **Consistency** for noise reduction. The general Consistency we proposed reflects the learning difficulty of the instance and provides guidance both in selecting soft targets of distillation and in weighting instances. Lastly, the generated instance weights directly multiply the sentence-level losses to dynamically and globally enhance the training in the sentence-level setting. The experiments on both held-out and manual datasets show that our approach boosts the student's performance to achieve state-of-the-art results and consistent improvements over the teacher. We also provide an ablation study to explore the effects of the modules. In addition, we analyse the errors and provide additional experimental results in the Appendix.

Our contributions are summarized as follows:

- We are the first to denoise sentence-level DSRE with dynamic instance weights and harness intermediate knowledge to improve noise resistance and information utilization.
- We propose a novel Transitive Instance Weighting mechanism with multiple functions, including noise alleviation, overfitting prevention, soft target selection and transitive knowledge passing.
- Experiment and analysis show that our method achieves state-of-the-art performance with good generalization and robustness.

2 Related Work

Distant supervision (DS) for relation extraction (Mintz et al., 2009) enables automatic annotation of large-scale datasets, but its strong assumption introduces a large number of wrongly labeled instances. Following Riedel et al. (2010), various multi-instance learning methods are proposed to denoise from noisy instances, and they broadly fall into two categories: instance selection (Zeng et al., 2015; Qin et al., 2018; Feng et al., 2018) and instance attention (Lin et al., 2016; Yuan et al., 2019b,a; Ye and Ling, 2019). Apart from multi-instance learning, many of the previous works try to improve the effectiveness of training. Liu et al. (2017) and Shang et al. (2020) try to convert wrongly labeled instances to useful information through relabeling. Huang and Du (2019) proposes collaborative curriculum learning for denoising. Hao et al. (2021) adopts adversarial training to filter noisy instances in the dataset. Nayak et al. (2021) designs a self-ensemble framework to filter noisy instances despite information loss. Li et al. (2022) proposes a hierarchical contrastive learning framework to reduce the effect of noise. Rathore et al. (2022) constructs a passage from the bags to generate a summary for classification. Nevertheless, the above approaches are trained with bag-level loss, leading to lower utilization of information. In our work, we adopt sentence-level training to directly utilize sentence-level information and effectively tackle noise and overfitting using dynamic instance weights.

Knowledge distillation (Hinton et al., 2015) is an effective way to improve model generalization, though it has difficulty in transferring knowledge effectively (Stanton et al., 2021). By sharing some parameters between teacher and students, self-distillation (Zhang et al., 2019a) improves knowledge transfer from teacher to student. Liu et al. (2020b) applies self-distillation on BERT (Devlin et al., 2019) to improve inference efficiency. However, in our work, we apply self-distillation as the tool to extract intermediate knowledge for denoising and further reduce the noise from the teacher with transitive information passing among the students.

There are some epoch-level techniques to detect noisy instances like Swayamdipta et al. (2020) and Huang et al. (2021). But in sentence-level DSRE which is highly noisy and contains bias from the entity mentions (Peng et al., 2020), larger models like BERT can overfit noisy instances faster, even before an epoch ends. Therefore, we adopt a dynamic instance weighting mechanism which is more suitable for DSRE.

3 Methodology

3.1 Overview

DSRE aims to predict the relations between an entity pair given a bag of sentences mentioning them. Previous works mostly aggregate sentence representations into bag representations before being optimized using bag-level loss. However, useful information may be diluted or mixed during aggregation. Instead, our model is directly trained on the sentence level to preserve more information.

Our model is illustrated in Figure 2. The backbone is the BERT encoder on the left, with a teacher classifier on the top. Each student shares a subencoder with the teacher and uses a new classifier for prediction. Firstly, the encoder and the teacher classifier are fine-tuned on the dataset to establish background knowledge. Then, we freeze the encoder and train the student classifiers, in which knowledge distillation and Transitive Instance Weighting (TIW) are applied to reduce noise and overfitting. TIW computes the instance weights based on three sources of knowledge: the teacher's output p^t, the outputs of the student i itself p_i^s and the previous peer p_{i-1}^s. It first selects a more consistent soft target p_i^{tg} between p^t and p_{i-1}^s based on the probabilities of making the same predictions as them (i.e. Consistency), which are denoted as c_i^t and c_i^s respectively. Then the possible false negative instances are filtered according to the predictions p_{i-1}^s from the previous peer. Finally, the instance weights w_i are computed as the multiplication of Uncertainty u_i (normalized entropy) and the general Consistency c_i with the soft target. The details of TIW are given in Algorithm 1, where $re2id(r)$ is a function that maps the relation class r to its id for generating the one-hot label. The instance weights directly multiply with the instance losses to dynamically regulate the roles of instances in optimization (Equation 6). This improves training by globally down-weighting the instances leading to extra noise and overfitting.

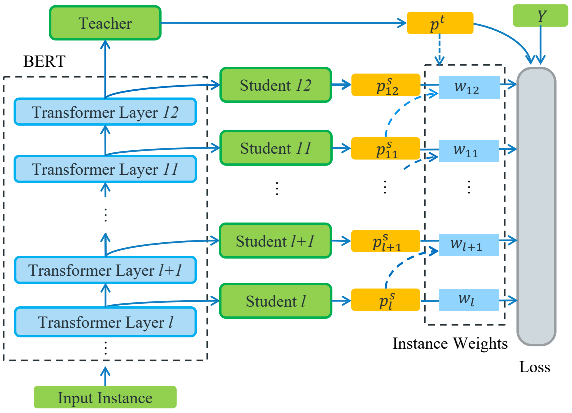

Figure 2: The overall framework of our model. Dotted arrows indicate the generation of instance weight.

3.2 Backbone

BERT (Devlin et al., 2019) is a powerful transformer-based pretrained network with broad applications in natural language processing. Its intermediate layers encode a rich hierarchy of sentence features, ranging from surface features, and syntactic features, to semantic features (Jawahar et al., 2019). However, previous BERT applications in DSRE (Alt et al., 2019; Rao et al., 2022) only utilize the output from the final layer, neglecting the possibility that hierarchical intermediate information can be useful in denoising. Therefore, we set up the student classifiers to extract information from the hierarchical features in the form of output probabilities and utilize them to distinguish noisy instances in the distillation stage.

The model takes a batch of sentences as input, each is labeled with at least one relation. Firstly, each input sentence is transformed into a sequence of vector representations s by the embedding layer. Then, BERT conducts layer-wise feature extraction with the input s, the output of i_{th} layer ($1 \leq i \leq n$) is described as:

$$h_i = BERT_i(s) \tag{1}$$

where $BERT_i$ refers to the subencoder containing transformer layers from the first to the i_{th}. The encoder is fine-tuned with a simple feedforward classifier FFN_t on the top and we can obtain the output of the teacher p^t as in the following:

$$x_i = [h_i(p_1); h_i(p_2)] \tag{2}$$

Algorithm 1 Transitive Instance Weighting

```
Input: DS label $Y$, teacher's output probability $p^t$ and students' $p^s$ for the instance.
Output: The soft target $p^{tg}$ and the instance weight $w$ of the instance from the students.
 1: Initialize $w_l \leftarrow 1, p_l^{tg} \leftarrow p^t$
 2: for $i = l + 1 \rightarrow n$ do
 3:     Compute the Consistency with teacher and peer: $c_i^t \leftarrow p_i^s \cdot p^t \quad c_i^s \leftarrow p_i^s \cdot p_{i-1}^s$
 4:     if $c_i^t > c_i^s$ then $p_i^{tg} \leftarrow p^t$ else $p_i^{tg} \leftarrow p_i^s$        ▷ Soft Target Selection
 5:     if $Y = rel2id(NA)$ then                                        ▷ False Negative Filtering
 6:         if $Y = argmax_j(p_{i-1}^s(j))$ then $w_i \leftarrow 1$ else $w_i \leftarrow 0$
 7:     else                                                            ▷ Positive Weighting
 8:         Compute the Uncertainty of soft target: $u_i \leftarrow \sum_{j=1}^{n_c} \frac{p_i^{tg}(j) log p_i^{tg}(j)}{log \frac{1}{n_c}}$
 9:         Compute the general Consistency and the instance weight: $c_i \leftarrow p_i^{tg} \cdot p_i^s \quad w_i \leftarrow c_i u_i$
10:     end if
11: end for
```

$$p^t = softmax(FFN_t(x_n)) \tag{3}$$

where p_1 and p_2 are the start positions of the head entity and tail entity respectively. $[a : b]$ indicates the concatenation of vectors a and b. n_c is the number of classes. Similarly, the output of student i can be formulated as follows:

$$p_i^s = softmax(FFN_i(x_i)) \tag{4}$$

Note that after fine-tuning, the parameters of the teacher model including the BERT encoder stay fixed during the process of self-distillation.

3.3 Transitive Instance Weighting

TIW incorporates multiple mechanisms to reduce noise and overfitting. For negative (NA) instances, TIW adopts **False Negative Filtering (FNF)** to filter false negatives based on the prediction of the previous peer p_{i-1}^s. For positive (non-NA) instances, TIW provides dynamic instance weights w_i generated by multiplying the **Uncertainty** u_i and the general Consistency c_i. The Uncertainty u_i is computed as the normalized entropy of the student's soft target p_i^{tg} as in Line 8 of Algorithm 1 and is applied to avoid overfitting to easy instances. The general Consistency c_i evaluates the consistency between the student's output and its soft target p_i^{tg} to limit the effects of wrongly-labeled instances.

Most previous works in knowledge distillation directly use the teacher's output probability as the soft target. However, the teacher can constantly make mistakes if trained with noisy data, as in DSRE. Therefore, by introducing peer output p^s into distillation, TIW sets up a transitive way to share knowledge among the students and reduces the noise from the teacher. As in Line 4 of Algorithm 1, instead of blindly following the output from the teacher, each student i $(i > l)$ chooses between the teacher p^t and the previous peer p_{i-1}^s to follow. This step is referred to as **Soft Target Selection (STS)** later. STS provides additional referential probability distributions for the learning students so they can switch to a smoother target probability when the output from the teacher is too hard to follow. The criterion of selection is Consistency (c_i^t and c_i^s), which is described as the probability of two systems making the same predictions and is computed as the dot product of the probability distributions from the two systems, as in Line 3 of Algorithm 1.

In TIW, we adopt different strategies for negative instances and positive ones because their characteristics are quite different. For negative instances, we conduct FNF as in Lines 5-6 of Algorithm 1. Since we have sufficient negative instances in the dataset, it is acceptable to avoid more FNs at the cost of slight information loss. Therefore, we assign 0 weight to all the possible FNs and 1 weight to the rest. To correctly identify FNs, we adopt a dynamic approach that if the previous peer agrees with distant supervision and also labels the instance as *NA*, then we classify the instance as a true negative. Otherwise, we assume it to be a false negative that the DS label is unreliable. The student follows the peer's view in FNF instead of the teacher's because the teacher already overfits the noisy data and mostly follows the DS label, though the probabilities of label relations may vary.

In order to preserve more information for training, we use soft weights for the positive instances instead of hard filtering. We call it **Positive Weighting (PW)** and determine the instance weight w_i of student i by two factors: Uncertainty u_i and the general Consistency c_i with the selected soft target.

The uncertainty term is the normalized entropy as in Liu et al. (2020b) of the chosen soft target. It evaluates how well an instance is fitted so we can leverage it to detect overfitted instances dynamically. Easy instances contain shallow features like *London, UK* indicating a *location/contains* relation, so the model fits them easily and fast. But we do not hope the model becomes overdependent on them and lose focus on deeper features hidden in semantics. Therefore we discount their weights with uncertainty to prevent overfitting.

The general Consistency c_i of student i is the Consistency between the student and the soft target. During distillation, each student is expected to stay consistent with its target distribution. If c_i is high, the student successfully follows the prediction of the soft target, indicating that the instance is easy to learn for the student. If c_i is low, the student fails to stay consistent with prior knowledge and the instance may be noisy or very hard to learn. The instance weight w_i should take the prevention of both noise and overfitting into consideration, so it is empirically implemented as the multiplication of general Consistency c_i and Uncertainty u_i, as in Line 9 of the algorithm.

Note that during distillation, the student is trained with both soft targets and DS labels, as shown in Equation 6. We present the discussions on the c_i and losses of easy, noisy and hard instances in the following.

Easy instances mostly have high c_i and are well-fitted by the teacher or the peer, so the optimizations using soft targets and DS labels conform with each other.

Noisy instances are mostly underfitted and very hard to optimize because the soft targets and DS labels are mostly inconsistent. They have low c_i because the teacher and the students are not likely to provide consistent predictions.

Hard instances are underfitted clean instances with low c_i at first. However, their soft targets and DS labels are consistent, leading to steady optimizations. When clean background knowledge is established by learning from clean instances, learning from hard ones becomes easier so the c_i values of hard instances grow larger.

Based on the above discussions, it is safe to say that both easy and hard instances are faster to fit than noisy ones during distillation, indicating that TIW is capable of reducing noise in the training set. As for Uncertainty, its role is non-decisive. Both hard and noisy instances tend to have high Uncertainty but the hard ones have higher Consistency, leading to larger weights than noisy ones. Easy instances are fast to fit even with their weights discounted by low Uncertainty. Therefore, applying Uncertainty helps alleviate overfitting and does not lead to increases in noise.

To sum up, TIW aims to tackle noise and overfitting and thus can be combined with sentence-level training, which is more demanding in both noise reduction and overfitting alleviation than traditional bag-level training.

3.4 Optimization

The teacher and the peer may overfit noisy instances during fine-tuning and distillation. Therefore, we apply a dynamic temperature τ to the soft target in the following form:

$$\tau_i = 1 + \gamma(1 - u_i) \tag{5}$$

where γ is a hyperparameter empirically set as 3. The idea of τ is to further smooth the well-fitted instances to produce softer targets.

The loss function of our model follows the general form of knowledge distillation with the instance weight w we propose:

$$L = \sum_{i=l}^{n} w_i(\alpha KL_{\tau_i}(p_i^s, p_i^{tg}) + (1-\alpha)CE(p_i^s, Y)) \tag{6}$$

where α is a hyper-parameter empirically set as 0.5. $KL_\tau(p, q)$ computes the KL-divergence between distributions p and q with temperature τ for the soft target q. Y is the label from distant supervision and $CE(p, Y)$ is the cross entropy loss with one-hot label obtained from Y.

4 Experiments

In this section, the datasets, settings and hyperparameters are specified first. Then, we present the performance of our model compared with previous baselines and the teacher model. We also conduct an ablation study to enable a deeper understanding of the mechanisms.

4.1 Datasets and Settings

We use the widely used **held-out** dataset NYT-10 (Riedel et al., 2010) and recent **manual** dataset NYT-10m (Gao et al., 2021) for evaluation. As a standard dataset for DSRE, NYT-10 is constructed by aligning the relations in Freebase (Bollacker et al., 2008) with the New York Times (NYT) corpus (English). NYT-10m is a manual dataset constructed also from the NYT corpus, with a human-labeled test set and a new relation ontology. For NYT-10, we divide the dataset into five parts for cross-validation. For NYT-10m, we use the provided validation set. The details of the datasets are shown in Table 1.

Dataset	Train (k)		Test (k)		Rel.
	Sen.	Fac.	Sen.	Fac.	
held-out	522.6	18.4	172.4	2.0	53
manual	417.9	17.1	9.7	3.9	25

Table 1: The statistics of datasets. **Sen.**, **Fac.** and **Rel.** indicate the numbers of sentences, relation facts and relation types (including *NA*) respectively.

In the experiments, we use the *bert-base-uncased* checkpoint with about 110M parameters for initialization as in Han et al. (2019). We apply the AdamW (Loshchilov and Hutter, 2017) optimizer during distillation. The structure of the embedding layer and BERT layers follow those in the previous works with the number of transformer layers $n = 12$ and hidden size $d_h = 768$. The batch size is 32 and the learning rate is $2e - 5$. The maximum length of sentences m is 128. As discussed by Jawahar et al. (2019), the shallow layers may not be able to encode the information needed for the DSRE task. Therefore, TIW starts from layer l, which is empirically set as 7.

We compare the Area Under precision-recall Curve (AUC), micro-F1, macro-F1, precision at top N predictions (P@N, N=100, 200, 300) and the mean of P@N, which is denoted as P@M. Following the *at-least-one* assumption (Riedel et al., 2010), we adopt **ONE** strategy (Zeng et al., 2015) for bag-level evaluation, which takes the maximum score for each relation to generate bag-level predictions. We use the output of the last student (12) as the output of the model.

In the Appendix, we display the results from other students and the results using other l settings. We also provide detailed error analysis and extra experimental results on Wiki-20m dataset. Moreover, we try out a different initialization of the teacher to further explore the generalization of TIW.

4.2 Overall Performance

We compare the performance of our model against that of the following baselines:

PCNN+ATT (Lin et al., 2016) proposes PCNN with selective attention mechanism.

RESIDE (Vashishth et al., 2018) integrates side information into Graph Convolution Networks to improve relation extraction.

DISTRE (Alt et al., 2019) extends and fine-tunes GPT on DSRE.

Intra+inter (Ye and Ling, 2019) combines intra-bag attention with inter-bag attention to tackle the noisy bags.

CIL (Chen et al., 2021) applies contrastive instance learning to reduce noise from DS.

HiCLRE (Li et al., 2022) uses a hierarchical contrastive learning Framework to improve DSRE.

PARE (Rathore et al., 2022) constructs a passage from the sentence bag and use its summary for relation extraction.

Teacher follows the implementation of Gao et al. (2021).

Among the baselines, DISTRE, HiCLRE, CIL and PARE use pretrained language models for initialization and the last three use the same BERT pretrained encoder as ours. The held-out dataset is the mainstream for DSRE evaluation, but it contains wrongly-labeled test instances leading to inaccurate evaluation. The manual dataset provides an accurate test set but is limited by its scale in generalization. Therefore, we use both datasets for better evaluation.

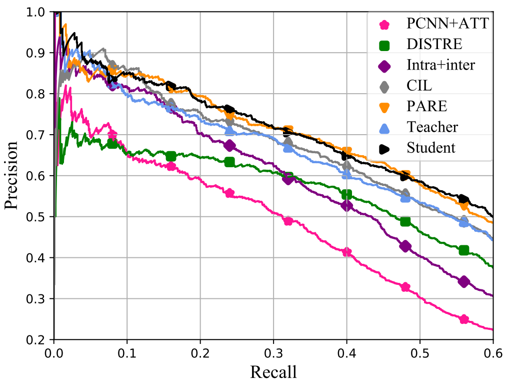

Figure 3: PR curves of the models on the held-out dataset.

Model	AUC	Micro-F1	P@100	P@200	P@300	P@M
PCNN+ATT	33.8	40.7	75.0	72.5	65.7	71.1
RESIDE	41.5	45.7	84.0	78.5	75.6	79.4
DISTRE	42.2	48.6	68.0	67.0	65.3	66.8
Intra+inter	42.3	46.5	**91.8**	84.0	78.7	84.8
CIL	50.8	52.2	90.1	**86.1**	81.8	**86.0**
HiCLRE	45.3	50.5	82.0	78.5	74.0	78.2
PARE	53.4	54.4	85.0	85.0	**82.7**	84.8
Teacher	50.6	52.2	90.0	82.0	78.7	83.6
Last Student	**53.9**	**55.3**	88.0	84.0	**82.7**	84.9

Table 2: The performance (%) of the models on the held-out dataset. The best scores are marked as **bold** and the second best scores are underlined, as in other tables of the experiments.

4.2.1 Evaluation on Held-out Dataset

Table 2 show the experimental results on the held-out dataset. We use the results reported in the papers of previous work. We also plot the precision-recall curves as in Figure 3.

As shown in the results, our model achieves the best AUC and Micro-F1 score among all the compared methods. It is shown that direct sentence-level training (the teacher) leads to a slight decline in precision due to the existence of noise but still achieves competitive AUC and Micro-F1 on the test set because of its advantage in information utilization. The P@N of the student are relatively lower than bag-level baselines, but still improved over the teacher. Compared with the teacher, the student further alleviates noise and overfitting with TIW, thus achieving state-of-the-art performance.

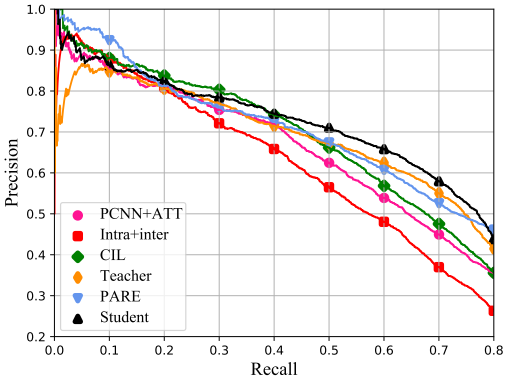

Figure 4: PR curves of the models on the manual dataset.

4.2.2 Evaluation on Manual Dataset

Table 3 shows the experimental results on the manual dataset. We use the original implementations of the baselines to reproduce the results. The precision-recall curves are plotted in Figure 4.

On the manual dataset, the bag-level methods still perform better at P@N, however, our method outperforms them in AUC and Micro-F1 by large margins. It shows that previous bag-level methods may overfit easy instances, leading to the loss of overall generalization despite higher precision at easy instances (P@N). Also, some of the baselines fail in handling infrequent relation classes (especially *Intra+inter*), but our model manages to achieve both high Micro-F1 and Macro-F1. Moreover, the student improves significantly over the teacher, especially in P@N. These results further demonstrate the effectiveness of TIW in improving sentence-level training.

According to Gao et al. (2021), the performance of the model may be inconsistent if evaluated in both the held-out and manual datasets. Good performance on the held-out set may indicate overfitting to the bias from DS. However, our model is robust enough to perform well on both datasets.

4.3 Ablation Study

The ablation study is performed using the held-out dataset. As shown in Table 4, all the modules improve the overall performance. Detailed discussions are given below:

a: removes Uncertainty and directly uses the general Consistency as positive weight. In this case, the easy instances always have the largest weights even if they are already well-fitted. The model thus overfits shallow features, which is indicated by the high P@N and the decline in overall performance.

b: removes STS and follows the output probabilities from the teacher all the time. In this case, the

Model	AUC	Micro-F1	Macro-F1	P@100	P@200	P@300	P@M
PCNN+ATT	57.7	57.0	21.2	91.0	88.5	88.0	89.2
Intra+inter	53.6	53.5	2.2	93.0	92.5	90.0	91.8
CIL	60.2	58.8	32.3	94.1	91.5	89.4	91.7
HiCLRE	61.8	62.8	34.7	85.0	84.5	83.7	84.4
PARE	62.7	60.7	**36.1**	**97.0**	**95.0**	**95.0**	**95.7**
Teacher	61.3	62.4	34.7	85.0	83.5	84.3	84.3
Last Student	**63.9**	**63.8**	35.2	94.0	90.5	88.0	90.8

Table 3: The performance (%) of our model and the baselines on the manual dataset.

Model	AUC	F1	P@M
Last Student	**53.9**	**55.3**	84.9
a: - Un	52.5	53.2	**86.1**
b: - STS	53.2	54.5	83.3
c: - PW	51.9	52.5	84.8
d: - FNF	53.3	54.9	82.5
e: - TIW	52.1	52.6	84.6
f: Probe	50.6	52.5	80.0

Table 4: Ablation study of our method.

noise from the teacher is not addressed. Fixing the soft target also leads to the fixed Uncertainty for each instance, causing the underfitting of some easy instances. Therefore, the performance declines, especially P@M.

c: removes PW and all the positive instances are treated equally, including the noisy ones. Therefore, the model is heavily affected by noise and FNF may be inaccurate, leading to further performance declines. In this case, high P@M indicates that the model overfits easy instances and loses generalization.

d: removes FNF. The FNs only make up a small part of the dataset, so the effects are relatively small. However, the noise from FNs significantly reduces P@M. We suspect that the fitting of FNs affects that of true positives. If a false negative fn has similar syntactic and semantic features to a true positive tp, fitting fn is similar to fitting tp using an incorrect label.

e: removes TIW totally and all the instances are weighted as 1. The label smoothness of knowledge distillation is able to alleviate some noise from DS, so there are improvements in performance over *f*. However, the student is still trained with much noise and overfits easy instances, so the overall performance declines significantly.

f: is the probing result of 12th layer using the DS label. It shows that without effective denoising mechanisms, simply retraining the classifier does not help in performance.

The above results and discussions further demonstrate the effectiveness of TIW designs in alleviating noise and overfitting.

5 Conclusions and Limitations

In this paper, we propose a novel Transitive Instance Weighting mechanism integrated with self-distillation to denoise from sentence-level training of DSRE. We employ the self-distilled BERT backbone to extract intermediate information for generating reliable instance weights. TIW combines Consistency with Uncertainty as the tools to tackle noisy instances and alleviate overfitting. It also enables soft target selection and transitive knowledge passing among the students to tackle the noise from the teacher. The experiment results show that our method improves the general resistance to DS noise and prevents overfitting from harming its generalization, thus can achieve state-of-the-art performance and consistent improvements over the baselines on both the held-out and manual datasets.

However, our work still has some limitations. Firstly, since our model is built on the basis of the teacher-student network, the performance of the student is highly affected by the teacher. If the teacher provides too much noisy information, our instance weighting mechanism might not work. Secondly, in some cases, the student fails to follow the correct predictions from the teacher, possibly due to ambiguity, lack of information or word-level noise. Finally, TIW may down-weight some instances of infrequent relation classes due to their difficulty, but it can be tackled by combining TIW with other methods addressing the long-tailed distribution of relations.

References

Christoph Alt, Marc Hübner, and Leonhard Hennig. 2019. Fine-tuning pre-trained transformer language models to distantly supervised relation extraction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1388–1398, Florence, Italy. Association for Computational Linguistics.

Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1247–1250.

Tao Chen, Haizhou Shi, Siliang Tang, Zhigang Chen, Fei Wu, and Yueting Zhuang. 2021. CIL: Contrastive instance learning framework for distantly supervised relation extraction. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6191–6200, Online. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Jun Feng, Minlie Huang, Li Zhao, Yang Yang, and Xiaoyan Zhu. 2018. Reinforcement learning for relation classification from noisy data. In *Proceedings of the aaai conference on artificial intelligence*, volume 32.

Tianyu Gao, Xu Han, Yuzhuo Bai, Keyue Qiu, Zhiyu Xie, Yankai Lin, Zhiyuan Liu, Peng Li, Maosong Sun, and Jie Zhou. 2021. Manual evaluation matters: Reviewing test protocols of distantly supervised relation extraction. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1306–1318, Online. Association for Computational Linguistics.

Xu Han, Tianyu Gao, Yuan Yao, Deming Ye, Zhiyuan Liu, and Maosong Sun. 2019. OpenNRE: An open and extensible toolkit for neural relation extraction. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP): System Demonstrations*, pages 169–174, Hong Kong, China. Association for Computational Linguistics.

Kailong Hao, Botao Yu, and Wei Hu. 2021. Knowing false negatives: An adversarial training method for distantly supervised relation extraction. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 9661–9672, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Geoffrey Hinton, Oriol Vinyals, Jeff Dean, et al. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2(7).

Xiusheng Huang, Yubo Chen, Shun Wu, Jun Zhao, Yuantao Xie, and Weijian Sun. 2021. Named entity recognition via noise aware training mechanism with data filter. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4791–4803, Online. Association for Computational Linguistics.

Yuyun Huang and Jinhua Du. 2019. Self-attention enhanced CNNs and collaborative curriculum learning for distantly supervised relation extraction. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 389–398, Hong Kong, China. Association for Computational Linguistics.

Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. 2019. What does bert learn about the structure of language? In *ACL 2019-57th Annual Meeting of the Association for Computational Linguistics*.

Dongyang Li, Taolin Zhang, Nan Hu, Chengyu Wang, and Xiaofeng He. 2022. HiCLRE: A hierarchical contrastive learning framework for distantly supervised relation extraction. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2567–2578, Dublin, Ireland. Association for Computational Linguistics.

Yankai Lin, Shiqi Shen, Zhiyuan Liu, Huanbo Luan, and Maosong Sun. 2016. Neural relation extraction with selective attention over instances. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2124–2133.

Tianyi Liu, Xiangyu Lin, Weijia Jia, Mingliang Zhou, and Wei Zhao. 2020a. Regularized attentive capsule network for overlapped relation extraction. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6388–6398, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Tianyi Liu, Xinsong Zhang, Wanhao Zhou, and Weijia Jia. 2018. Neural relation extraction via inner-sentence noise reduction and transfer learning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2195–2204.

Tianyu Liu, Kexiang Wang, Baobao Chang, and Zhifang Sui. 2017. A soft-label method for noise-tolerant distantly supervised relation extraction. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1790–1795.

Weijie Liu, Peng Zhou, Zhiruo Wang, Zhe Zhao, Haotang Deng, and Qi Ju. 2020b. FastBERT: a self-distilling BERT with adaptive inference time. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6035–6044, Online. Association for Computational Linguistics.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Mike Mintz, Steven Bills, Rion Snow, and Dan Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 1003–1011.

Tapas Nayak, Navonil Majumder, and Soujanya Poria. 2021. Improving distantly supervised relation extraction with self-ensemble noise filtering. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2021)*, pages 1031–1039, Held Online. INCOMA Ltd.

Hao Peng, Tianyu Gao, Xu Han, Yankai Lin, Peng Li, Zhiyuan Liu, Maosong Sun, and Jie Zhou. 2020. Learning from Context or Names? An Empirical Study on Neural Relation Extraction. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3661–3672, Online. Association for Computational Linguistics.

Pengda Qin, Weiran Xu, and William Yang Wang. 2018. Robust distant supervision relation extraction via deep reinforcement learning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2137–2147.

Ziqin Rao, Fangxiang Feng, Ruifan Li, and Xiaojie Wang. 2022. A simple model for distantly supervised relation extraction. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 2651–2657, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Vipul Rathore, Kartikeya Badola, Parag Singla, and Mausam . 2022. PARE: A simple and strong baseline for monolingual and multilingual distantly supervised relation extraction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 340–354, Dublin, Ireland. Association for Computational Linguistics.

Sebastian Riedel, Limin Yao, and Andrew McCallum. 2010. Modeling relations and their mentions without labeled text. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 148–163. Springer.

Yuming Shang, He-Yan Huang, Xian-Ling Mao, Xin Sun, and Wei Wei. 2020. Are noisy sentences useless for distant supervised relation extraction? In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8799–8806.

Wei Song, Weishuai Gu, Fuxin Zhu, and Soon Cheol Park. 2023. Interaction-and-response network for distantly supervised relation extraction. *IEEE Transactions on Neural Networks and Learning Systems*.

Samuel Stanton, Pavel Izmailov, Polina Kirichenko, Alexander A Alemi, and Andrew G Wilson. 2021. Does knowledge distillation really work? *Advances in Neural Information Processing Systems*, 34:6906–6919.

Swabha Swayamdipta, Roy Schwartz, Nicholas Lourie, Yizhong Wang, Hannaneh Hajishirzi, Noah A. Smith, and Yejin Choi. 2020. Dataset cartography: Mapping and diagnosing datasets with training dynamics. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9275–9293, Online. Association for Computational Linguistics.

Shikhar Vashishth, Rishabh Joshi, Sai Suman Prayaga, Chiranjib Bhattacharyya, and Partha Talukdar. 2018. Reside: Improving distantly-supervised neural relation extraction using side information. *arXiv preprint arXiv:1812.04361*.

Zhi-Xiu Ye and Zhen-Hua Ling. 2019. Distant supervision relation extraction with intra-bag and inter-bag attentions. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2810–2819.

Changsen Yuan, Heyan Huang, Chong Feng, Xiao Liu, and Xiaochi Wei. 2019a. Distant supervision for relation extraction with linear attenuation simulation and non-iid relevance embedding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 7418–7425.

Yujin Yuan, Liyuan Liu, Siliang Tang, Zhongfei Zhang, Yueting Zhuang, Shiliang Pu, Fei Wu, and Xiang Ren. 2019b. Cross-relation cross-bag attention for distantly-supervised relation extraction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 419–426.

Daojian Zeng, Kang Liu, Yubo Chen, and Jun Zhao. 2015. Distant supervision for relation extraction via piecewise convolutional neural networks. In *Proceedings of the 2015 conference on empirical methods in natural language processing*, pages 1753–1762.

Linfeng Zhang, Jiebo Song, Anni Gao, Jingwei Chen, Chenglong Bao, and Kaisheng Ma. 2019a. Be your own teacher: Improve the performance of convolutional neural networks via self distillation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3713–3722.

Xinsong Zhang, Pengshuai Li, Weijia Jia, and Hai Zhao. 2019b. Multi-labeled relation extraction with attentive capsule network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 7484–7491.

Peng Zhou, Wei Shi, Jun Tian, Zhenyu Qi, Bingchen Li, Hongwei Hao, and Bo Xu. 2016. Attention-based bidirectional long short-term memory networks for relation classification. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 207–212.

A Hyperparameter Analysis

There are two key hyperparameters in our experiments, the student selected and the head layer l. In our best model, we select the last student (12th) for evaluation and set layer 7 as the head layer.

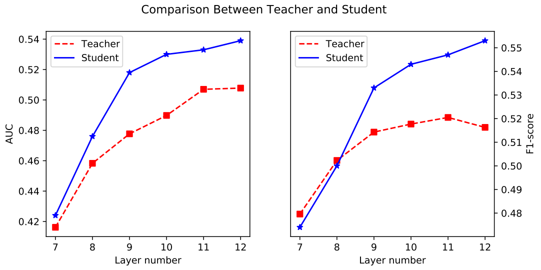

Figure 5: Results of the students and auxiliary classifiers of the teacher on the held-out dataset.

As shown in Figure 5, the higher students(≥ 9) improve significantly over the teacher. The last student performs the best and the students from 9th to 11th also achieve comparable performances. Lower layers of BERT encode shallower features and the instance weighting in lower students is more affected by noise, so the performances of 7th and 8th students show little advantage over the teacher. With knowledge passed and noise alleviated student by student, the performance gradually improves.

To study the effect of head layer l, we run experiments with l from 1 to n. In Table 5, we present the results where $l = 7$ achieves the best performance. For $l > 7$, the head layer is too close to the top and TIW filters fewer false negatives. So the P@M declines quickly, which is similar to the effect of removing FNF as in Table 4. For $l < 7$, the lower layers of BERT are not able to encode sufficient information for accurate relation extraction, so the lower students are not able to provide reliable instance weights, leading to the transfer of some noise among students. Though other settings are less effective than the best, their performances still dominate most of the baselines. The above results show that our method is not dependent on the empirical settings of hyperparameters and further demonstrate the effectiveness and robustness of our method.

Setting	AUC	F1	P@M
$l = 11$	53.4	55.1	82.8
$l = 10$	53.5	54.9	83.6
$l = 9$	53.6	55.0	84.0
$l = 8$	53.7	55.1	84.7
$l = 7$	**53.9**	**55.3**	**84.9**
$l = 6$	53.8	**55.3**	84.8
$l = 5$	53.7	55.1	84.6
$l = 3$	53.5	55.0	84.7
$l = 2$	53.5	54.9	84.6
$l = 1$	53.4	54.9	84.5

Table 5: Results of using different head layer l settings. The best results are marked as **bold**.

B Evaluations on Wiki-20m

In order to further explore the generalization of our method, we also experiment on the Wiki-20m dataset (Gao et al., 2021). The details of the dataset are shown in Table 6 and the results are shown in Table 7.

Train (k)		Test (k)		Rel.
Sen.	Fac.	Sen.	Fac.	
698.7	157.7	138.0	56.0	81

Table 6: The statistics of Wiki-20m dataset. **Sen.**, **Fac.** and **Rel.** indicate the numbers of sentences, relation facts and relation types (including *NA*) respectively.

Model	AUC	Macro-F1
Intra+inter	88.7	81.1
CIL	89.7	82.6
HiCLRE	87.9	80.3
IRN (Song et al., 2023)	<u>90.9</u>	82.5
PARE	**91.4**	<u>83.9</u>
Teacher	89.7	82.8
Last Student	<u>90.9</u>	**84.1**

Table 7: The performance (%) of our model and the baselines on the Wiki-20m dataset.

Sentence	Teacher	Student
Carl Friedrich von Weizsäcker was born in *Kiel*, Germany, on June 28, 1912.	/people/person/place_of_birth	/people/person/place_lived
Presented by *Brooklyn College* and the office of Borough President *Marty Markowitz*.	/business/person/company	/people/person/place_lived
Furthermore, the relationship between the central government, dominated by three small *Arab* tribes living along the Nile, and Darfur's Arabs, who claim a heritage going back to the Prophet *Muhammad*, is often antagonistic.	/people/person/ethnicity	/people/person/place_of_birth

Figure 6: *TCSI* examples. The entities are underlined.

We take the best-reported results of the baselines in Rathore et al. (2022) and Song et al. (2023). On Wiki-20m dataset, our model still achieves state-of-the-art performance and the improvements over the teacher are significant. Therefore, our method can generalize well to the Wiki-20m dataset, which has more relation classes (81).

C Error Analysis

For accurate analysis of the errors, we use the test set of the manual dataset for statistical discussions. Each positive label is considered an **item**. The instances with multiple positive labels are considered to have multiple items. We classify the items based on the predictions of the teacher and student, then count the number and percentage of each class as in Table 8. The goal is to explore where the errors of the student come from: a) **from the teacher**, meaning that the knowledge from the teacher is noisy and leads to the student's errors, or b) **from the student itself**, meaning that the teacher gives correct knowledge but the student fails to follow.

Class	Num. of items	Percentage (%)
BC	3,044	78.07
BI	742	19.03
TISC	94	2.41
TCSI	19	0.49

Table 8: Numbers and percentages of different classes of items. *BC* stands for *both correct*, *BI* stands for *both incorrect*, *TISC* stands for *teacher incorrect, student correct* and *TCSI* stands for *teacher correct, student incorrect*.

In the results, the student achieves slightly higher (about 2%) accuracy than the teacher and shows high fidelity with 97.1% of all predictions being the same as the teacher. *BI* represents the student's errors caused by the errors from the teacher. *TISC* indicates the student's corrections on the errors from the teacher and *TCSI* represents the errors from the student itself. From the results, we can conclude that almost all (about 97.5%) of the errors come from the teacher, and the corrections made by the student are much more than the errors made by the student itself. This demonstrates the effectiveness of our method in reducing the occurrence of errors and the limitation that it requires a good teacher for good performance.

For further analysis of the student's errors, we inspect the *TCSI* items and select some representative ones for discussions as in Figure 6. Most of the instances with *place_of_birth* relation are correctly classified and the first example should be an easy instance in the form, yet misclassified by the student as *place_lived*. We observe several similar items and suspect that long and uncommon names like *Carl Friedrich von Weizsäcker* sometimes confuse the student to make conservative predictions, which is the more common relation *place_lived*. The second example, however, confuses the student with a compound noun *Brooklyn College*. *Brooklyn* appears very often in the dataset in the form of location, making the student believe that *Brooklyn College* is a location rather than an organization. The third example is mostly related to ambiguity, where the word *Arab* may refer to the Arab people (ethnic group) or the Arab world (location). The latter two examples indicate that the lack of entity-related information may lead to inconsistency between the student and the teacher. The first example shows that the student may be confused to lose focus on key phrases like *was born in*, which may be solved by combining it with word-level attention in the future.

D Effects of Teacher Model

In the main experiments of this paper, the teacher model is trained in an extremely noisy environment, leaving much room for TIW to improve performance. In order to explore the potential of TIW in improving state-of-the-art methods, we initialize the teacher model using the well-trained encoder from CIL (Chen et al., 2021) instead of the *bert-base-uncased* checkpoint. We repeat the experi-

Model	AUC	Micro-F1	P@100	P@200	P@300	P@M
CIL	60.2	58.8	94.1	91.5	89.4	91.7
PARE	62.7	60.7	**97.0**	95.0	95.0	95.7
Teacher	61.3	62.4	85.0	83.5	84.3	84.3
Teacher†	61.7	60.6	89.0	90.0	88.7	89.2
Last Student	**63.9**	**63.8**	94.0	90.5	88.0	90.8
Last Student†	63.4	60.7	96.0	**96.0**	**95.3**	**95.8**

Table 9: The performance (%) of the models on the manual dataset. † indicates initialized using well-trained encoder.

ments on the manual dataset and the results are shown in Table 9. In the results, the models initialized with a well-trained CIL encoder achieve significantly higher precision and TIW further improves the performance over the baselines. However, since CIL is trained in a bag-level setting, it has lower utilization in information than the models trained in sentence-level settings, leading to some decline in AUC and Micro-F1. These results show that the initialization of the teacher model has a great impact on the performance and that TIW can consistently improve the performance of the model with different teachers. Hopefully, TIW can be employed with more powerful teacher models to achieve even better performance in DSRE.