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 for 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.

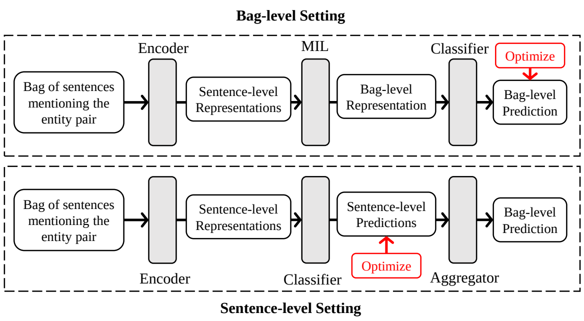

Figure 1: The bag-level and sentence-level pipelines of DSRE.

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) becomes popular in recent years, the information within the outputs from their intermediate layers is a non-trivial source of knowledge but is rarely utilized 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 easy instances to improve the learning of deeper knowledge. To tackle the above challenges, we propose a novel **Transitive Instance Weighting (TIW)** mechanism for DSRE. Our method adopts the sentence-level setting in both stages: fine-tuning and distillation. After fine-tuning the BERT encoder using a linear classifier (teacher) in the first stage, we append an auxiliary classifier (student) to each relevant layer and train them with TIW during distillation. TIW first filters FNs using binary weights (0 or 1). Then the soft target probabilities are chosen between the outputs of the teacher and the previous peer. Finally, the instance weights for the positive (non-NA) instances are generated by combining two factors: the uncertainty (Liu et al., 2020b) and the soft confidence score. We apply uncertainty to prevent overfitting easy instances and use the soft confidence score as the assessment of learning difficulty, where easy and hard instances tend to have higher scores than noisy ones. During filtering and weighting, each student receives information from both the teacher and the previous peer, enabling the alleviation of noise from the teacher and transitive knowledge passing among the students. The experiments on both held-out and manual datasets show that our approach achieves state-of-the-art performance and consistent improvements over the teacher and the baselines. We also provide a detailed ablation study to explore the effects of the modules. Finally, we analyse the errors and discuss the limitations of our method.

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. 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. 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 students. 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

Our model is illustrated in Figure 2. The backbone of our model is the BERT encoder on the left, with a teacher classifier on the top. Each student contains a subencoder and an auxiliary classifier. For example, the student 7 has a subencoder ending with the 7th BERT layer and a linear classifier appended to it. The BERT encoder is fine-tuned with the teacher classifier on the dataset before distillation. As discussed in 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 and will be called **the head layer** in the rest of the paper.

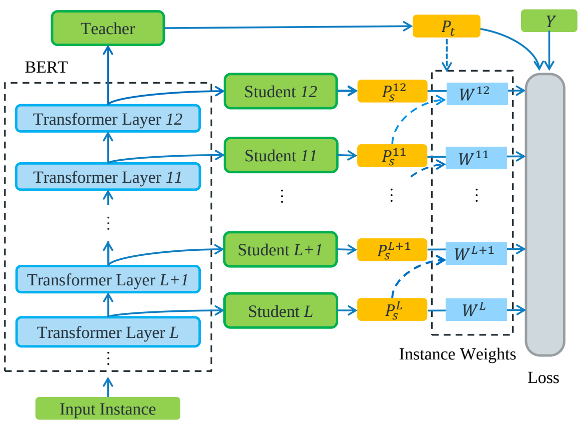

Figure 2: The overall framework of our model. Dotted arrows indicate the generation of instance weight.

3.1 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 apply auxiliary classifiers as in Figure 2 to extract information from the hierarchical features in the form of output probabilities and utilize them to distinguish noisy instances in the distillation stage.

Before distillation, we fine-tune the BERT encoder on DSRE as in Gao et al. (2021). 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 $d_h = 768$.

Firstly, the input sentence is transformed to a sequence of vector representations $s = [w_1, w_2, ..., w_m]$ by the embedding layer, where m is the maximum length of the sentence. 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 on the top:

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

$$FFN(h_i) = M_2(M_1 x_i + b_1) + b_2 \tag{3}$$

$$p^t = softmax(FFN_t(h_n)) \tag{4}$$

where $M_1 \in R^{d_h \times d_h}$ and $M_2 \in R^{n_c \times d_h}$ are weight matrices and $b_1 \in R^{d_h}$ and $b_2 \in R^{n_c}$ are bias terms. 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. x_i is the entity-aware sentence representation generated by concatenating the hidden vectors of the entity pair. n_c is the number of classes and p^t is the output probability of the teacher.

The student i can be formulated as follows:

$$p_i^s = softmax(FFN_i(h_i)) \tag{5}$$

After fine-tuning, the parameters of the teacher model including the BERT encoder stay fixed.

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 **PoA**s of i_{th} student: $c_i^t \leftarrow p_i^s \cdot p^t \quad c_i^s \leftarrow p_i^s \cdot p_{i-1}^s$
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 instance weight: $w_i \leftarrow max(c_i^t, c_i^s) u_i$
10: **end if**
11: **end for**

3.2 Transitive Instance Weighting

The algorithm of TIW is shown in Algorithm 1, where $re2id(r)$ is a function that maps the relation class r to its id for generating the one-hot label. TIW provides dynamic instance weights for each student except the first one (layer L), it sets up a transitive way to share knowledge (output probabilities) among the students. Note that we use the last student for the final prediction and the rest of the students aim to provide robust instance weights for the last one.

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, as in Line 4 of our algorithm, instead of blindly following the output from the teacher, each student except the first one chooses between the teacher p^t and the previous peer p_{i-1}^s to follow. The criterion of choosing is consistency, which can be described as the probability of making the same predictions as each other. We call it the **Probability of Agreement (PoA)** and compute it as the dot product of two probability distributions. The selection of soft targets provides additional referential probability distributions for the learning students and they can switch to a smoother target probability when the output from the teacher is too hard to learn.

In TIW, we adopt different strategies for negative (NA) instances and positive (non-NA) ones because their characteristics are quite different. We conduct **False Negative Filtering (FNF)** as in Lines 5-6 of the algorithm. 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 and the soft confidence score.

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 <u>*London, UK*</u> indicating a <u>*location/contains*</u> 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 maximum between the PoAs from the teacher and the previous peer is the **Soft Confidence (SC)** score which evaluates the learning difficulty of the instance for the student. If the SC score is high, the student successfully follows the idea of the teacher or the peer, indicating that the instance is easy to learn for the student. If the SC score is low, the student is unable to follow the referential probabilities and the instance may be noisy or very hard to learn.

The instance weight for i_{th} student ($l < i \leq n$) is computed as the product of the SC score and the uncertainty term, as in Line 9 of the algorithm. Note that during distillation, the student is trained with both soft target distribution and DS labels, as shown in Equation 7. We present the discussions on the SC scores and losses of easy, noisy and hard instances in the following.

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

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

Hard instances are underfitted clean instances with low SC scores at first. However, their soft and hard labels are consistent, leading to smoother optimizations. When clean background knowledge is established by learning from clean instances, learning from hard ones becomes easier so the SC scores 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 and tend to have larger SC scores than noisy ones. The uncertainty term only takes effect when easy instances are well-fitted and clean background knowledge is established, so it will not lead to overfitting noisy instances.

To sum up, TIW is robust against noise and overfitting and thus can be combined with sentence-level training to utilize more information for better performance than previous bag-level methods.

3.3 Optimization

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

$$\tau_i = 1 + \gamma(1 - u_i) \tag{6}$$

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:

$$L = \sum_{i=l}^{n} w_i(\alpha KL_{\tau_i}(p_i^s, p_i^{tg}) + (1-\alpha)CE(p_i^s, Y)) \tag{7}$$

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 teacher. 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 and error analysis to enable a deeper understanding of the mechanisms.

4.1 Datasets and Settings

We use two datasets for evaluation, the widely used **held-out** dataset NYT-10 (Riedel et al., 2010) and recent **manual** dataset NYT-10m (Gao et al., 2021). 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). The training set includes sentences from 2005 to 2006, and the test set uses sentences from 2007. NYT-10m is a manual dataset constructed also from 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 and fix the random seed as 42. Apart from the hyperparameters previously mentioned, the batch size is 32 and the learning rate is $2e-5$. The maximum length of sentences m is 128. The head layer L is set as layer 7 in our experiments.

We compare the Area Under precision-recall Curve (AUC), the F1 score and the mean of precision at top N predictions (N=100, 200, 300), 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. As mentioned in Section 3, We use the output probabilities of the last student as the output of our model. In the appendix, we also display the results from other students and the results using other settings of L.

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 finetunes 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.

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

Among the baselines, DISTRE and CIL use pretrained language models for initialization. CIL adopts the same BERT pretrained encode 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 of the datasets for better evaluation.

4.2.1 Evaluation on Held-out Dataset

Table 2 shows 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.

Model	AUC	F1	P@M
PCNN+ATT	33.8	40.7	71.1
RESIDE	41.5	45.7	79.4
DISTRE	42.2	48.6	66.8
Intra+inter	42.3	46.5	84.8
CIL	<u>50.8</u>	<u>52.2</u>	**86.0**
Teacher	50.6	<u>52.2</u>	83.6
Student	**53.9**	**55.3**	<u>84.9</u>

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 <u>underlined</u>, as in other tables of the experiments.

As shown in the results, our model achieves the best AUC and F1 score among all the compared methods. The P@M of the student is relatively lower than bag-level methods, but still significantly higher than the teacher model. We can see that sentence-level training leads to a slight decline in the P@M due to the existence of noisy sentences but achieves better overall performance on the test set because of its advantage in information utilization. Our method further alleviates noise and overfitting with TIW, thus achieving state-of-the-art performance.

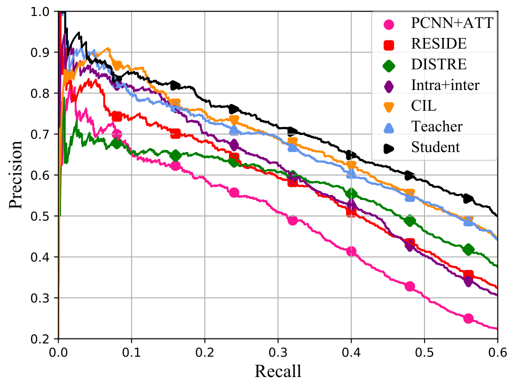

Figure 3: PR curves of the models on the held-out dataset.

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. The precision-recall curves are plotted in Figure 4.

In the results, the bag-level methods still perform better at P@M, however, our method outperforms them in AUC and F1 by large margins.

Model	AUC	F1	P@M
PCNN+ATT	57.7	57.0	89.2
Intra+inter	53.6	53.5	**91.8**
CIL	60.2	58.8	91.7
Teacher	61.3	62.4	84.3
Student	**63.9**	**63.8**	90.8

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

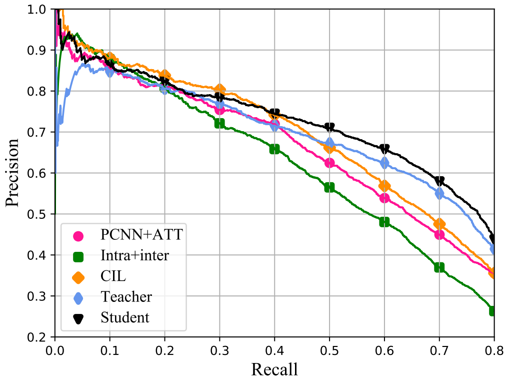

Figure 4: PR curves of the models on the manual dataset.

It shows that previous bag-level methods overfit easy instances, leading to the loss of overall generalization. The student also achieves significant improvements over the teacher, especially in P@M. The 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

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

Table 4: Ablation study of our method.

As shown in Table 4, all the modules improve the overall performance. Detailed discussions are given below:

a: removes Soft Target Selection (STS) and follows the output probabilities from the teacher all the time. The noise from the teacher is not addressed, leading to performance declines.

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

c: removes FNF. The false negative instances only make up a small part of the dataset, so the effect is relatively small. However, the noise from FNs significantly reduces P@M. We suspect that the fitting of false negatives 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.

d: 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 *e*. However, the student is still trained with much noise and overfits easy instances, so the overall performance declines significantly.

e: 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.

4.4 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 5. 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.

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 5: *TCSI* examples. The entities are underlined.

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

Table 5: 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 5. 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 with word-level attention in the future.

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 the soft confidence score with uncertainty 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 due to ambiguity, lack of information or word-level noise, which indicates that further extension of our method is plausible. Finally, we haven't explored other instance weighting methods in this paper. There might be better solutions yet to be discovered.

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.

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.

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.

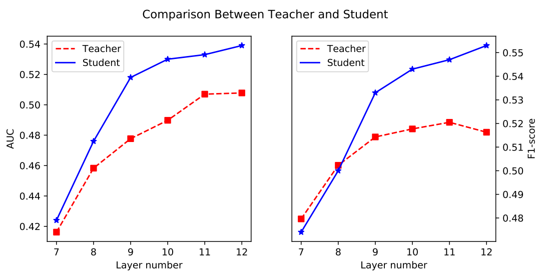

Figure 6: PR curves of the students and auxiliary classifiers of the teacher on the held-out dataset.

As shown in Figure 6, 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.

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 6: Results of using different head layer L settings. The best results are marked as **bold**.

To study the effect of head layer L, we run experiments with L from 1 to n. In Table 6, 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 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.