Self-Attention Enhanced CNNs and Collaborative Curriculum Learning for Distantly Supervised Relation Extraction

Yuyun Huang[†] **Jinhua Du**[‡§]
[†]University College Dublin, Ireland
[‡]Investments AI, AIG (American International Group, Inc.)
[§]ADAPT Centre, School of Computing, Dublin City University, Ireland
{yuyun.huang}@ucd.ie
{jinhua.du}@aig.com

Abstract

Distance supervision is widely used in relation extraction tasks, particularly when large-scale manual annotations are virtually impossible to conduct. Although Distantly Supervised Relation Extraction (DSRE) benefits from automatic labelling, it suffers from serious mislabelling issues, i.e. some or all of the instances for an entity pair (head and tail entities) do not express the labelled relation. In this paper, we propose a novel model that employs a collaborative curriculum learning framework to reduce the effects of mislabelled data. Specifically, we firstly propose an internal self-attention mechanism between the convolution operations in convolutional neural networks (CNNs) to learn a better sentence representation from the noisy inputs. Then we define two sentence selection models as two relation extractors in order to collaboratively learn and regularise each other under a curriculum scheme to alleviate noisy effects, where the curriculum could be constructed by conflicts or small loss. Finally, experiments are conducted on a widely-used public dataset and the results indicate that the proposed model significantly outperforms baselines including the state-of-the-art in terms of P@N and PR curve metrics, thus evidencing its capability of reducing noisy effects for DSRE.

1 Introduction

Relation Extraction (RE) is vital for NLP tasks such as information extraction, question answering and knowledge base completion. RE aims to identify the relationship between an entity pair (e_1, e_2) in a sentence. For example, in the sentence "$[Bill\_Gates_{e_1}]$ is the principal founder of $[Microsoft_{e_2}]$", the relation extractor decodes the relation of *founder* for the entity pair *Bill_Gates* (the person) and *Microsoft* (the company).

Recent supervised relation extraction research can be roughly categorised into two areas: fully supervised and distantly supervised relation extraction. Fully supervised relation extraction mainly depends on manually annotated training dataset (Zeng et al., 2014; dos Santos et al., 2015). Ordinarily, human annotation on large-scale datasets is costly and often practicably impossible. Distance supervision addresses this problem by using an existing knowledge base (e.g. DBpedia) to automatically annotating large-scale datasets, thus reducing the burden of manual annotation (Mintz et al., 2009; Hoffmann et al., 2011; Surdeanu et al., 2012). However, distance supervision often suffers from mislabelling problems.

Figure 1 illustrates this incorrect labelling issue, which shows that not all sentences in a bag with the same entity pair express the labelled relation of *person/company/founder*. The worst case is that all sentences in a bag are mislabelled. Thus, one primary challenge in DSRE is to minimize the noisy labelling effects, which in turn, would let model learn from incorrect labelled datasets.

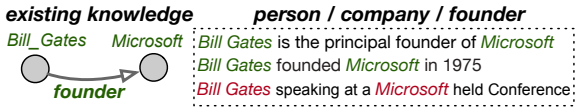


Figure 1: Mislabeling issue example in DSRE: the entity pairs bag containing three sentences is labeled as *person/company/founder*. However, the last sentence marked in red has no extractable pre-defined relation.

In order to design a solution to mitigate the effects of noisy data, we can treat our DSRE model learning procedure as analogous to *two students* training with a *curriculum* to answer a list of multiple-choice questions, where some difficult multiple-choice questions may have no correct answers (false positive). We base our proposed solution on the intuition that two students will compare and rethink their different answers during the learning process, thus regularising each

Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, pages 389–398, Hong Kong, China, November 3–7, 2019.

other and improving their final grades. We call this process of two students learning together with a curriculum as *Collaborative Curriculum Learning* (CCL), where each student represents a network.

Inspired by the above intuition, we propose to use a bag-level selective sentence attention model (NetAtt) (Lin et al., 2016) and a maximum probability sentence model (NetMax) (Zeng et al., 2015) as two students learning collaboratively. Meanwhile, highly organised human education methods using a multilevel curriculum, from easy to hard, is an analogy for curriculum learning training, which advocates the adoption of similar multi-stage strategies (Bengio et al., 2009). The curriculum learning training method for DSRE is used with the assumption that entity pair bags contain corrupted labelled sentences, which are difficult components to learn in the curriculum. By disregarding the effects of noisy samples (meaningless knowledge) during the training, the expectation is to boost the model's learning capability. Moreover, in order to accurately obtain the semantic representation of each sentence for our curriculum learning approach, we propose an internal CNNs self-attention mechanism to learn a better sentence representation in the DSRE setting.

The main contributions are summarised as:

(1) We make the first attempt to use the concept of curriculum learning for denoising DSRE and present a novel collaborative curriculum learning model to alleviate the effects of noisy sentences in an entity pair bag. In this model, we define two collaborative relation extractors to regularize each other and boost the model's learning capability.

(2) We propose conflicts and small loss tricks for our collaborative curriculum learning. Instead of using a separated complex noisy sentence filter and two-step training in baseline models, our model can alleviate noise effects during a single training and is easy to implement.

(3) We are the first to apply an internal CNNs self-attention mechanism to enhance a multilayer CNNs model for DSRE.

(4) We conduct thorough experiments on the widely-used NYT dataset, and achieve significant improvements over state-of-the-art models.

2 Related Work

Most DSRE approaches fall under the framework of Multi-Instance Learning (MIL) (Riedel et al., 2010; Surdeanu et al., 2012; Zeng et al., 2015; Lin et al., 2016; Ji et al., 2017; Qin et al., 2018; Feng et al., 2018). At the encoding step, a sentence representation is learned using handcrafted features or neural network models. Afterwards, in the sentence selection step, one or several sentences from an entity pair bag are chosen for further bag representation learning. Previously, statistical models (Mintz et al., 2009; Hoffmann et al., 2011; Surdeanu et al., 2012) have used designed features, such as syntactic and lexical features, and have then been trained by logistic regression or expectation maximization.

When adopting deep learning approaches, a single layer CNNs based model (Zeng et al., 2014) was exploited to extract sentence level features to attain fully supervised relation classification. For DSRE, Zeng et al. (2015) proposed an extended Piece-wise CNN (PCNN) approach and selected the most probable valid sentence to represent an entity pair bag, while the remaining sentences in the bag were ignored. Lin et al. (2016) and Ji et al. (2017) used all the sentences in a bag by assigning higher weights to valid labeled sentences and lower weights to noisy sentences. The selective sentence attention mechanism combined all weighted sentences as a bag representation. In addition, Ji et al. (2017) made use of entity description background knowledge and fused the external information into their PCNN-based model.

The self-attention mechanism (Cheng et al., 2016; Parikh et al., 2016; Vaswani et al., 2017), also called intra-attention, relates to different positions of a single sequence to learn the sequence representation. An internal CNNs states self-attention approach was proposed by Zhang et al. (2018) to improve Generative Adversarial Networks (GANs) (Goodfellow et al., 2014) performance in generating high-quality images. Alt et al. (2019) extended Generative Pre-trained Transformer (GPT) to learn semantic and syntactic features for DSRE. Wang et al. (2019) used pre-trained Transformer for multiple entity-relations extraction task. Du et al. (2018) utilized self-attention mechanisms for better MIL sentence-level and bag-level representations. However, previous work has not considered to use self-attention over the internal CNNs model states for DSRE.

Deeper CNNs have positive effects on noisy NLP tasks (Conneau et al., 2017). Huang and Wang (2017) used residual learning for multilayer deep CNNs to improve DSRE performance.

To better address issues relating to mislabelling, e.g., when *some or all sentence labels* in a bag are falsely positive, schemes to filter out noisy instances have been developed. Takamatsu et al. (2012) proposed a wrong label sentence filter method using linguistic features. Feng et al. (2018) proposed a model comprising of a relation classifier and an instance selector based on reinforcement learning (RL). The instance selector was designed to select possible correctly labelled sentences and was regularised by the rewards from the relation classifier. Qin et al. (2018) used the idea of GANs to develop a separated correct label indicator, which filters high confidence scoring instances for training on existing PCNN/CNN-based relation extractors (DSGAN). *Unlike previous work*, which filters out incorrectly labelled *sentences* to generate an approximate clean training dataset and then retrains on the filtered data to improve models, we instead train our model to actively and purposefully forget noisy entity pair *bags*, based on a collaborative curriculum learning strategy in a single training process. In doing so, we develop an approach to building the curriculum – the identified disagreements and losses of two collaborative student networks in our model.

3 Methodology

The mislabelled sentences from the distance supervision method are normally regarded as unwanted noise that can reduce the performance of relation extraction. To alleviate noisy effects, we propose a collaborative curriculum learning framework for DSRE. The *architecture* is shown in Figure 3, consisting of three main components: (1) input representation; (2) CNNs that are composed of convolution, self-attention and pooling; and (3) collaborative curriculum learning module.

3.1 Inputs: Word and Position Embeddings

To represent each input token, we use word2vec[1] (Mikolov et al., 2013) to obtain its embedding. Each word embedding (w_t) contains syntactic and semantic information.

Figure 2: relative distances from 'founder' to entities

Similar to (Zeng et al., 2015), we use position embedding to assist the CNNs in measuring distances from the current word to the head and tail entities. As illustrated in Figure 2, in the sentence, the distance of the word *founder* to the head entity is 4 and -2 to the tail entity.

Figure 3(a) further illustrates the use of these embeddings as input representation in CNNs, i.e. the input representation is a concatenation of *word* embedding and *position* embedding. In Figure 3(a), it is assumed that the dimensions of word embedding d_w and position embedding d_p are 3 and 1 (as a simplified example for the sake of the figure), respectively. The total vector representation dimension d is $d_w + 2 \times d_p$.

3.2 Contextualised Representation: Self-attention for CNNs

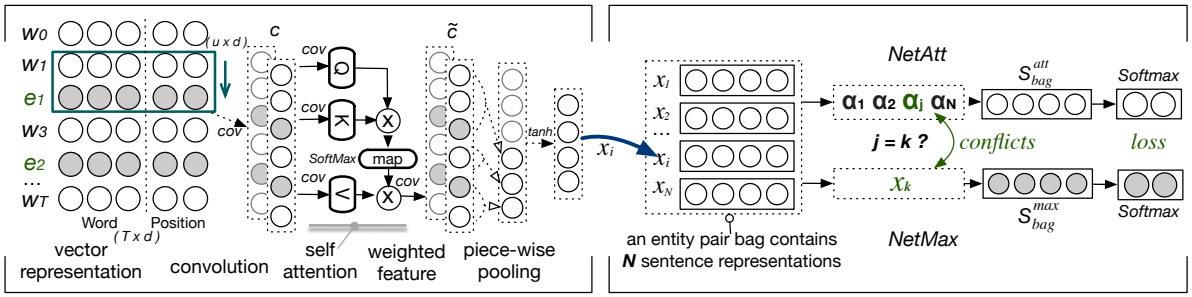
The sentence embedding matrix is formed by concatenating every vector representation horizontally (left panel Figure 3) and is represented as $s_n = [w_1 : \cdots : w_t : \cdots : w_T]$, where s_n is the input that feeds CNNs to learn a sentence representation. For an input sentence, we use a convolution filter W_p to slide along s_n as $[w_t : \cdots : w_{t+u-1}] * W_p + b$, where $*$ is the convolution symbol, $b \in \mathbb{R}$ is a bias and p represents the p^{th} filter in a filter set. $W_p \subset \mathbb{R}^{u \times d}$, where u is the length of the filter and d is the dimension of a word vector consisting of word and position embeddings.

In order to learn a better sentence representation, we propose a self-attention mechanism which is performed directly over internal CNNs states as shown in Figure 3(a). The self-attention function maps queries (Q) and corresponding keys (K) to compute a weight map. The output is a multiplication of the values (V) and the weight vector. We place this internal CNNs self-attention module after the first convolution state (C). Values, queries, and keys are computed by applying convolution again on C, where $V = C * W^V$, $Q = C * W^Q$ and $K = C * W^K$. The attention map, is calculated using the softmax function, as shown in Equation (1), where Q^T is the transpose of Q and $\otimes$ is the matrix multiplication operator.

$$map = \frac{\exp(Q^T \otimes K)}{\sum \exp(Q^T \otimes K)} \quad (1)$$

Subsequently, the weighted value is computed as $\tau V \otimes map$. $\tilde{C} = cov(C + \tau V \otimes map)$ is then fed into a piece-wise max pooling layer. Where, τ learns gradually from 0 to 1 to assign more weight to the model.

[1] https://code.google.com/p/word2vec/



(a) Self-Attention Piece-wise pooling CNN *(b) Collaborative Learning with Conflicts and Loss*

Figure 3: ***(a)*** A sentence embedding matrix with the entity pair (e_1, e_2) consists of a word vector set of dimension $(T \times d)$. T is the length of the given sentence and d is the length of the word and position vector. A convolution filter with a dimension of $(u \times d)$ is sliding along the sentence representation. cov represents a convolution operation. The internal CNNs states (C) from the first convolution operation are fed into the self-attention module. A piece-wise max-pooling is employed at outputs $(\tilde{C})$ from last convolution layer. A sentence representation x_i is learned after a nonlinear function. ***(b)*** For an entity pair bag B containing N sentence representations, the j^{th} sentence with maximum weight score α_j is selected by NetAtt, while x_k is the k^{th} sentence selected by NetMax. Conflicts between the two subnets are used to form a conflict loss L_{jk}. Each network uses the same sentence representation and generates different bag representations to feed a softmax layer separately. The *conflicts* and *loss* from collaborative training are used as cues for curriculum building in section 3.5.2.

3.3 Entity Position-aware Sentence Representation: Piece-wise Max Pooling

Following the self-attention operations, we use a piece-wise max pooling operation to form the final sentence representation. The max pooling is a variant of pooling in standard CNNs, which applies the pooling to three convolution segments separated by head and tail entities (Zeng et al., 2015). As shown in the column of "piece-wise pooling" in Figure 3(a), the grey dots in $\tilde{c}$, representing head and tail entities, split each vector into three pieces, which are denoted as $\tilde{C}_1, \tilde{C}_2, \tilde{C}_3$, respectively. Thus, the piece-wise max pooling is expressed as $\{x_i^1, x_i^2, x_i^3\} = \text{max-pooling}(\tilde{C}_1, \tilde{C}_2, \tilde{C}_3)$. These three pooled vectors are then concatenated together to form a vector x_i and a nonlinear function is applied to the output vector, such that $x_i = tanh(x_i)$ is the final representation of a sentence.

3.4 Bag Representation for Entity Pairs

After learning representations of all sentences in an entity pair bag, we then use two approaches to form the entity pair bag representation, as illustrated in Figure 3(b), i.e., we employ a mechanism of averaging all sentences using attention scores (NetAtt) (Lin et al., 2016), and a maximum probability sentence selection method (NetMax) (Zeng et al., 2015), respectively, to learn the bag representation. Specifically, NetAtt assigns weight scores $\{\alpha_1, \ldots, \alpha_N\}$ to all sentences in an entity pair bag, while NetMax will select the most reasonable sentence that has the highest probability.

Ideal student networks are equipped with SOTA MIL selection mechanisms and could empirically generate conflicts during the selection, based on this criterion we use NetAtt and NetMax. NetAtt works by considering all sentences in a bag, but it also introduces noisy sentences while learning a bag representation. NetMax works by selecting the sentence with the highest probability in a bag as the bag representation, but it overlooks other valid sentences. Moreover, the two networks use different bag representations to feed classifiers and eventually generate *disagreements*. Therefore, we are motivated by the idea of combining their advantages and disagreements over sentence selection in a single framework to learn better bag representations and to reduce the noise effects.

Before introducing the proposed single framework, we first describe below how NetAtt and NetMax work in bag representation learning. The sentence bag of an entity pair is denoted as B, which consists of representations of N sentences $\{x_1, x_2, \ldots, x_N\}$, where each sentence representation is learned from our self-attention based PCNN. The entity pair is expressed as (e_1, e_2) and the bag's relation is r.

3.4.1 NetAtt: Sentence-Level Attention

To extract information from all sentences in a bag, a sentence-level attention mechanism is used to learn a weight score α_i for each sentence. Subse-

quently, the bag representation S^{att}_{bag} for the entity pair's bag B is computed as: $S^{att}_{bag} = \sum_{i=1}^{N} \alpha_i x_i$.

We can see that the purpose of the weighted factor α_i is to give higher weights to correctly labeled instances and lower weights to wrongly labeled ones. Given the score β_i of a given sentence representation x_i with a relation r, which is measured as: $\beta_i(x_i|r) = x_i A r$ (where A is a weighted diagonal matrix), the attention scores in a given bag B can be calculated as: $\alpha_i(x_i|r, B) = softmax(\gamma\beta_i(x_i|r))$, where γ is set empirically and borrowed from the work by (Sutton and Barto, 1998). Smaller γ will lead to equalisation in all sentences and larger γ will increase bias of the high scored sentence.

3.4.2 NetMax: Maximum Probability

NetMax assumes that at least one sentence in an entity pair bag reflects the bag's relation, and only one sentence with the maximum probability is selected to represent the bag, which is denoted as $S^{max}_{bag} = x_k$, where x_k is the k^{th} sentence representation with maximum probability. As only one sentence is selected, the input o for the *softmax* classifier can be expressed as $o = Kx_k + b$, where K is the transformation matrix and b is the bias. For a bag with relation r, the conditional probability is $p(r|x_k;\theta) = softmax(o_r)$. For all sentences in a bag, the index k is computed by $k = arg\max_k p(r|x_k;\theta)$.

3.5 Collaborative Curriculum Learning

As mentioned above, we consider the advantages and disagreements of sentence selection of NetAtt and NetMax in a single framework so that they can learn to regularise each other so as to reduce the effects of noisy sentences. We propose a collaborative curriculum learning framework where NetAtt and NetMax are defined as two student networks and they learn together under a curriculum scheme. For DSRE, we assume that entity pair bags with wrongly labeled sentences are *hard samples* to be learned, while bags with correctly labeled sentences are *easy samples*. Figure 3(b) shows the architecture of our collaborative curriculum learning framework, where NetAtt and NetMax are trained collaboratively and regularised by each other. The curriculum vector v_i for collaborative learning could be built by various schemes, for example, the *conflicts* (v^c_i) of selecting the valid sentence in a bag between the two student networks, which will be detailed later.

3.5.1 Objective Function

The objective function is defined as $J(S_i;\theta) = \frac{1}{m}\sum_{i=1}^{m} j(S_i;\theta)$, where m is the total number of entity pair's bags in a mini-batch. S_i is a set of $\{S^{att}_{bag\_i}, S^{max}_{bag\_i}\}$. $j(S_i;\theta)$ is the objective function of one entity pair's bag defined in Equation (2):

$$\begin{aligned} j(S_i;\theta) = \eta \log p(r_i|S^{att}_{bag\_i};\theta) + \\ (1-\eta)\log p(r_i|S^{max}_{bag\_i};\theta) \end{aligned} \quad (2)$$

where, $0 < \eta < 1$ is an empirical value to assign weights to NetAtt and NetMax. With a curriculum, the model's minimisation problem can be formulated as in Equation (3):

$$\min_{\theta,v} E(\theta, v, \lambda) = \frac{1}{m}\sum_{i=1}^{m} v_i l_i + L_{jk} \quad (3)$$

where, $l_i = l(r_i, j(S_i;\theta))$ is the loss of each bag; r_i is the relation of a bag; v_i is the curriculum weight variable; L_{jk} is the cross entropy loss of conflicts between the highest probability sentence indexes from NetAtt and NetMax, which aims to let them regularise each other during sentence selection; θ and λ are the optimisation parameters of relation extractor and curriculum. The weighted loss is minimised by stochastic gradient descent.

Algorithm 1 Update using conflicts

inputs: mini batch size m; j^i_{att} and k^i_{max}; two students: p_{att} (*NetAtt*), p_{max} (*NetMax*); bag representation: S^{att}_i, S^{max}_i

for *i in* {*1,2,3, . . . , m*} **do**
  if $j^i_{att} = k^i_{max}$ **then**
    # *no conflicts* $v^c_i = 1$
    add S^{att}_i to set $\{S^{att}\}$
    add S^{max}_i to set $\{S^{max}\}$
  end
end
update p_{att} *with* $\{S^{att}\}$ **&** p_{max} *with* $\{S^{max}\}$

3.5.2 Curriculum Construction for Collaborative Learning

Conflicts trick of two students are utlised to build a curriculum (v^c_i): In a mini-batch of size m, where the i^{th} entity pair's bag contains N sentences, j^i_{att}, $k^i_{max} \in [0\ to\ N)$ are the indexes of highest probably sentence selected by NetAtt and NetMax, respectively. For each entity pair bag in the batch, if j^i_{att} is not equal to k^i_{max}, then $v^c_i = 0$, representing a *'hard' sample* (with conflicts). Otherwise $v^c_i = 1$, representing an *'easy' sample*. The

conflicts that occur during the collaborative training between NetAtt and NetMax are shown in Figure 3(b). There is no extra curriculum network ($f(v^c;\lambda)$) required to learn a v_i^c.

When v_i^c is assigned to 0, the training procedure will forget the effects of *'hard'* sample (i^{th} bag) by multiplying 0 and its loss l_i as shown in Eq. (3). Algorithm 1 illustrates the logic to update the training using the conflicts curriculum. In a mini-batch, entity pair bags with conflicts in sentence selection are dropped, the remaining bags are used to update the network parameters.

Furthermore, various curriculum types could be used in CCL to alleviate the noise in DSRE. We utilize the **small loss trick** to build a curriculum which inspired by Jiang et al. (2018). Specifically, to build a curriculum (v_i^l), the loss (l_i) is used as an input constant feature to learn a curriculum vector. We use the MentorNet framework[2] (Jiang et al., 2018) as the curriculum network to learn an approximate predefined curriculum $f(v^l;\lambda)$. The approximation process is to train a curriculum model using synthetic data generated according to the predefined curriculum. The trained model is then used as the curriculum to guide the further model training. We use the predefined curriculum $f(v^l;\lambda) = v_i^l l_i + \frac{1}{2}\lambda_2 (v_i^l)^2 - (\lambda_1 + \lambda_2) v_i^l$ (Jiang et al., 2015) to guide our training.

4 Experiments

4.1 Dataset

We evaluate our model on the widely used New York Times (NYT) DSRE dataset[3], which aligns Freebase entity relation with NYT corpus (Riedel et al., 2010). The dataset uses the data from 2005 to 2006 as the training set and the remaining data, taken from 2007, as the test set. The processed dataset[4] was released by Lin et al. (2016). We use the cleaned version of the processed dataset, which has removed duplicated sentences in training and test sets. In total, the training set consists of *522,611* sentences, 281,270 entity pairs and 18,252 relation facts. The testing set contains 172,448 sentences, 96,678 entity pairs, and 1,950 relation facts. The dataset contains 39,528 unique entities and 53 relations in total including an *NA* relation which represents no existing relation for given entity pairs in sentences.

[2] https://github.com/google/mentornet
[3] http://iesl.cs.umass.edu/riedel/ecml/
[4] https://github.com/thunlp/NRE

4.2 Evaluation Metrics

Instead of obtaining a costly human annotated test data, we conduct a held-out evaluation (Riedel et al., 2010) in the experiments, as in previous work (Riedel et al., 2010; Surdeanu et al., 2012; Zeng et al., 2015; Lin et al., 2016). Held-out evaluation compares relation facts predicted in test data with those relations identified in Freebase. It gives an approximate evaluation of the proposed model. As with the evaluation metrics used in the literature, we report our results using Precision-Recall curve (PR-curve) and Precision at N (P@N) metrics. *PR-curve* is used to understand the trade-off between precision and recall. Using all the test data, the plotted curve expresses the precision as a function of recall. *P@N* considers the cutoff top-most *N* precision values as a set. Each entity pair bag contains one or more instances and P@N considers *all* the multiple instances.

4.3 Baseline Models

We compare our model with both statistical and deep learning baseline models. These baselines were evaluated on the same cleaned dataset. We exclude some recent models due to the dataset and reproducibility issues. Namely, recent models that were trained on a dataset, which was released by mistake, obtained higher results. These results in fact may be inaccurate. From the Github repository commit history and comments[5]: *"It has not deleted the mix part of testing data. The training sentence is 570000+, but in the paper is 520000+."*. The incorrect training dataset was replaced with the corrected one in March 2018, which might be the main reason as to why some MIL DSRE papers in 2018 reported non-reproducible results on the corrected training data. In all probability these works had commenced prior to March 2018 and were likely relying on the erroneous dataset.

(1) Statistical Models: *Mintz* (Mintz et al., 2009) extracted sentence syntactic and lexical features and trained a multiclass logistic regression classifier. *Hoffmann* (Hoffmann et al., 2011) is a probabilistic, graphic model with MIL. *MIMLRE* (Surdeanu et al., 2012) is a MIL model that uses expectation maximization for classification.

(2) Deep Learning Models: both CNN (Zeng et al., 2014) and PCNN (Zeng et al., 2015) based

[5] https://github.com/thunlp/NRE/commit/77025e5cc6b42bc1adf3ec46835101d162013659

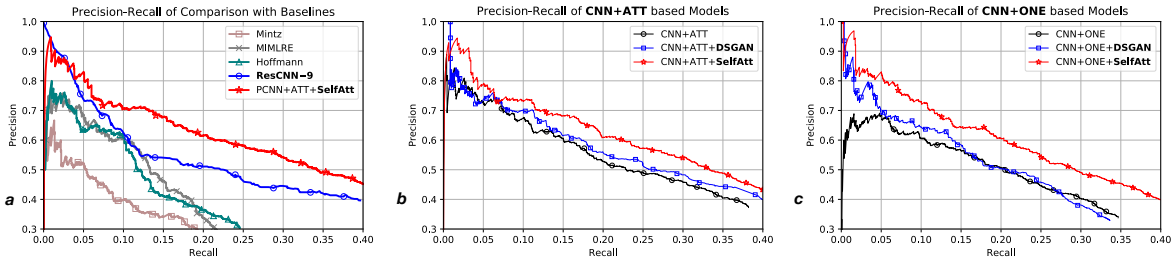


Figure 4: PR curves comparison of CNN/PCNN based models with SelfAtt module and baselines

relation extraction models with bag level selective attention mechanism (*PCNN+ATT*, *CNN+ATT*) (Lin et al., 2016) and multi-instance learning (*PCNN+ONE*, *CNN+ONE*) (Zeng et al., 2015).

(3) ResCNN-9 (Huang and Wang, 2017): 9-layers CNN model with residual identity shortcuts and three fully connected layers. The model outperforms CNN+ATT/ONE models.

(4) State-of-the-art DSRE Noise Filter Systems: DSGAN (Qin et al., 2018) is a model to filter out noise instances. It unitizes GANs to remove potentially inaccurate sentences from original training data and further trains PCNN/CNN models on the filtered data to improve performance.

Number of CNN filters	128, 230, 256
Batch size B	120
Learning rate	0.1, 0.4
Weight decay	0.00001
Burn-in epoch	5
Dropout probability	0.3

Table 1: Parameters setting for best results

We fine-tune our models by validating and selecting the best model parameters. To accomplish this we set the gradient descent learning rate among $\{0.4, 0.2, 0.1, 0.01, 0.001\}$. Batch size B is set to $\{60, 120, 160\}$. The amount of CNN filter is set among $\{64, 128, 230, 256\}$. A dropout rate is in the set $\{0.1, 0.3\}$. All the parameters fine-tuned in our experiments are shown in Table 1. Specifically, the best performance of [CNN+ATT/ONE+SelfAtt] with training parameters of learning rate:0.4 and CNN filters:230; the best performance of [PCNN+ATT/ONE+SelfAtt] with training parameters of learning rate:0.4 and CNN filters:128; and the best performance of [PCNN+ATT/ONE+SelfAtt+CCL] with training parameters of learning rate:0.1 and CNN filters:256. For other parameters, we follow the settings in the work of (Lin et al., 2016), e.g. the maximum sentence length is limited to 70, the word and position embedding size is fixed to 50 and 5 respectively, and the CNNs filter window size is 3.

4.4 Effects of Internal CNNs Self-Attention

Armed by the internal multilayer CNNs self-attention mechanism (named as **SelfAtt**), our model can learn a better representation from noisy data compared with the conventional CNN encoder used in PCNN/CNN. Each model has more than one convolution operation, as illustrated in Figure 3(a). Prior to feeding these outputs from the first convolution operation into the self-attention module, we add a batch normalisation followed by an ReLU activation function.

Figures 4 & 5 show the PR-curve results attained by applying SelfAtt to CNN/PCNN based models. We also report the results of P@100, P@200, P@300 and the Mean for CNN/PCNN+ONE and CNN/PCNN+ATT with SelfAtt in the held-out evaluation. Table 3 shows the P@N values with test settings where all sentences in an entity pair's bag are taken into account. PR-curve and P@N results demonstrate that CNN/PCNN based approaches achieve improved results with the SelfAtt. PCNN/CNN models with SelfAtt also outperform ResCNN-9 in terms of P@N and PR-curve, which indicates that the proposed SelfAtt is beneficial for boosting the performance of the models learning from noisy inputs. The state-of-the-art DSGAN system demonstrates its ability to improve PCNN/CNN + ATT/ONE by filtering out noisy data. By comparing our SelfAtt with DSGAN, Figure 4 shows that SelfAtt significantly outperforms the DSGAN system in terms of CNN-based models.

The intuition of adding an internal CNN states self-attention module to help DSRE task is that, 1/ a deeper CNN has positive effects on noisy NLP tasks (Conneau et al., 2017), 2/ attention enhanced PCNN/CNN is expected to assign various weight scores to different sentence portions and will form a better representation in the DSRE setting.

By applying the SelfAtt it could add more

convolution layers into a model, as the internal self-attention used is placed between two convolution operations. Huang and Wang (2017) demonstrated that multi-layer ResCNNs network does achieve performance improvement by adding residual identity shortcuts, which aligns with the study that deeper CNN has positive effects on noisy NLP tasks (Conneau et al., 2017). However, previous DSRE researches overlooked multilayer CNN/PCNN with ONE/ATT. To investigate multilayer effects of DSRE, we present multilayer CNN with ATT results in Table 2, we observe that with ATT several multilayer models (e.g., 2 layers CNN+ATT) have improvements compared with single layer models, and overfitting occurs when larger convolution layers are employed.

P@N(%)	100	200	300	Mean	AUC
CNN-1+ATT	76.2	68.6	59.8	68.2	0.327
CNN-2+ATT	76.2	**72.1**	**68.4**	**72.2**	**0.348**
CNN-5+ATT	74.2	**72.6**	**66.1**	**71.0**	**0.338**
CNN-9+ATT	66.3	64.2	63.7	64.7	0.315

Table 2: Multilayer CNNs+ATT

Thus, our SelfAtt design is expected to boost a sentence encoder with attention scores and alleviate noise effects by benefiting from a multilayer CNN network. From empirical testing, to apply attention more subtly, placing self-attention after the second convolution operation and followed by one convolution operation works well for all PCNN/CNN + ONE/ATT based models generally. We report results with this setting.

avaya , which was once a division of lucent technologies and att before that , is one of the nation 's top makers of phone equipment , rivaling cisco , nortel and alcatel-lucent in providing internet-based communications to corporations .

By looking at the weight scores from SeftAtt, we observe that different parts of a sentence obtain different attentions. For example, as the heat-map illustrates using attention scores, entities *avaya* and *cisco* obtain more attention than others.

4.5 Effects of Collaborative Curriculum Learning

We refer to the Collaborative Curriculum Learning using Conflicts Trick as CCL-CT, and using the Small Loss trick as CCL-SL. For the following experiments, we expect that by applying our CCL strategies, it will result in further improvements by reducing the undesirable effects of noise. In our framework of integrating collaborative curriculum learning, both NetAtt and NetMax could be used for testing. We report a comparison of NetAtt/NetMax+SelfAtt+CCL, PCNN+ONE/ATT and state-of-the-art DSGAN noise reduction as shown in Table 3 and Figure 5.

P@N (%)	100	200	300	Mean
CNN-based Models				
CNN+ONE	67.3	64.7	58.1	63.4
ResCNN-9	79.0	69.0	61.0	69.7
CNN+ONE+**SelfAtt**	**81.1**	**75.1**	**70.4**	**75.5**
CNN+ATT	76.2	68.6	59.8	68.2
CNN+ATT+**SelfAtt**	**81.1**	**74.1**	**72.4**	**75.9**
PCNN-based Models				
PCNN+ONE	72.3	69.7	64.1	68.7
PCNN+ONE+**SelAtt**	84.1	75.1	69.1	**76.1**
[NetMax+**SelfAtt**]+**CCL-CT**	**85.1**	**78.6**	**74.4**	**79.4**
PCNN+ATT	76.2	73.1	67.4	72.2
PCNN+ATT+**SelfAtt**	81.1	71.6	70.4	**74.4**
[NetAtt+**SelfAtt**]+**CCL-CT**	**82.2**	**79.1**	**73.1**	**78.1**

Table 3: P@N results for models with internal CNNs self-attention and curriculum learning

CCL-CT utilises the conflicts from the collaborative training of NetMax and NetAtt to form a curriculum to guide the training. The two students form different bag representations to feed to Softmax layer separately, and they generate disagreements on the bag-level selection during training. In our experiment, each epoch reveals less than 10.9% disagreement, thus we drop less than 10.9% of the total entity pair bags during each epoch and the drop ratios work well in our experiments.

Model		**+SelfAtt**	**CCL-CT**
PCNN+ATT	0.341	**0.368**	**0.381**
PCNN+ONE	0.325	**0.352**	**0.380**

Table 4: Comparison of AUC Results

From Figures 5(a) and 5(b), we can see that the CCL based models have further improvements in terms of PR-curves compared with PCNN+ATT/ONE+SelfAtt. The P@N results in Table 3 indicate that CCL further improves the model's performance when compared to PCNN+ATT/ONE+SelfAtt as well. Table 4 gives another comparison using AUC with all p-values less than 5e-02 from t-test evaluation. The results indicate that the larger AUC, the better performance. A simple ensemble model of two networks (AUC: 0.371) has a similar result as a single model (NetAtt, AUC: 0.368). The main purpose of adding an additional student network is to introduce conflicts and to build the collaborative curriculum learning scheme. With the CCL strategies, our models improve performances by removing *'hard'* (noisy) entity pair bags during

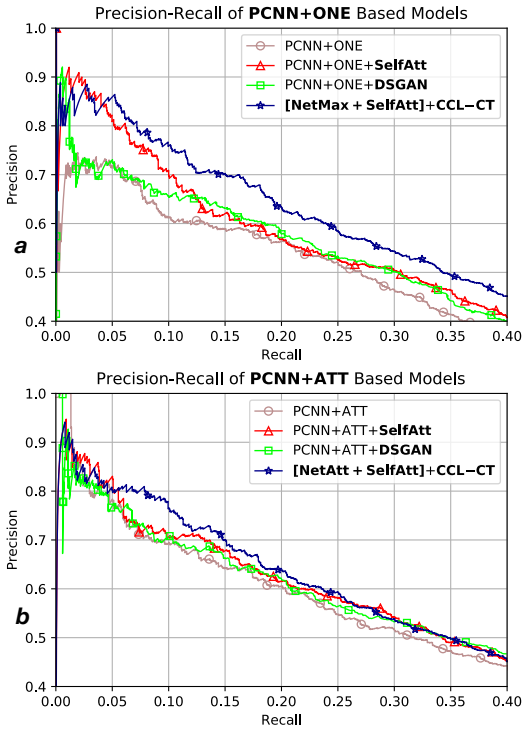


Figure 5: PR curves comparison of PCNN based noise removal models

training. When compared with the state-of-the-art DSGAN system, our models outperform both DSGAN based ONE and ATT models.

We examine the small loss trick based curriculum using MentorNet (Jiang et al., 2018). *CCL-SL* uses the loss from the collaborative training to build the curriculum vector, which commences by guiding the collaborative training from the burn-in epoch (5^{th}) and the optimal epoch result of (AUC:0.382, P@N mean:77.3%), which is reported for held-out evaluation starting at the burn-in epoch. The results also demonstrate that various curriculum types (the conflict and loss tricks) could help to alleviate the noise in DSRE.

Overall, our experimental results demonstrate that the proposed SelfAtt and CCL strategies for PCNN/CNN models significantly outperform baselines in terms of PR-curve and P@N.

5 Conclusion

To deal with the mislabelling issue in distantly supervised relation extraction, this paper details the development of a novel model based on a multi-layer self-attention mechanism for CNNs and collaborative curriculum learning strategies with two students (NetAtt and NetMax). The internal self-attention model can learn a better sentence representation by taking advantage of deeper CNNs in terms of positive effects on noisy inputs. The CCL strategies can perform a collaborative training on NetAtt and NetMax by allowing them to regularize each other, in tandem with the removal of noisy sample effects. Two different tricks, namely conflicts tricks and small loss tricks, are utilized in the CCL framework. Experimental results on the commonly-used NYT dataset indicate that our proposed approaches significantly outperform state-of-the-art baseline models in terms of P@N and PR-curve evaluation metrics.

6 Acknowledgement

We thank co-authors Jingguang Han and Sha Liu ({jingguang.han, sha.liu}@ucd.ie) and anonymous reviewers for these insightful comments and suggestions. We would like to thank Emer Gilmartin (gilmare@tcd.ie) for helpful comments and presentation improvements. This research is funded by the Enterprise-Ireland Innovation Partnership Programme (Grant IP2017626).

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. ACL.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. 2009. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pages 41–48. ACM.

Jianpeng Cheng, Li Dong, and Mirella Lapata. 2016. Long short-term memory-networks for machine reading. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 551–561.

Alexis Conneau, Holger Schwenk, Loïc Barrault, and Yann Lecun. 2017. Very deep convolutional networks for text classification. In *European Chapter of the Association for Computational Linguistics EACL'17*.

Jinhua Du, Jingguang Han, Andy Way, and Dadong Wan. 2018. Multi-level structured self-attentions for distantly supervised relation extraction. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 2216–2225.

Jun Feng, Minlie Huang, Li Zhao, Yang Yang, and Xiaoyan Zhu. 2018. Reinforcement learning for relation classification from noisy data. In *Proceedings of AAAI*.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *Advances in Neural Information Processing Systems 27*, pages 2672–2680.

Raphael Hoffmann, Congle Zhang, Xiao Ling, Luke Zettlemoyer, and Daniel S Weld. 2011. Knowledge-based weak supervision for information extraction of overlapping relations. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 541–550. ACL.

YiYao Huang and William Yang Wang. 2017. Deep residual learning for weakly-supervised relation extraction. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1803–1807.

Guoliang Ji, Kang Liu, Shizhu He, and Jun Zhao. 2017. Distant supervision for relation extraction with sentence-level attention and entity descriptions. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA.*, pages 3060–3066.

Lu Jiang, Deyu Meng, Qian Zhao, Shiguang Shan, and Alexander Hauptmann. 2015. Self-paced curriculum learning. In *AAAI*.

Lu Jiang, Zhengyuan Zhou, Thomas Leung, Li-Jia Li, and Li Fei-Fei. 2018. Mentornet: Learning data-driven curriculum for very deep neural networks on corrupted labels. In *ICML*.

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)*, volume 1, pages 2124–2133.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 3111–3119. Curran Associates, Inc.

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 ACL and the 4th IJCNLP of the AFNLP*, pages 1003–1011. Association for Computational Linguistics.

Ankur Parikh, Oscar Täckström, Dipanjan Das, and Jakob Uszkoreit. 2016. A decomposable attention model for natural language inference. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2249–2255.

Pengda Qin, Weiran Xu, and William Yang Wang. 2018. Dsgan: Generative adversarial training for distant supervision relation extraction. *arXiv preprint arXiv:1805.09929*.

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.

Cícero Nogueira dos Santos, Bing Xiang, and Bowen Zhou. 2015. Classifying relations by ranking with convolutional neural networks. In *ACL*. Association for Computational Linguistics.

Mihai Surdeanu, Julie Tibshirani, Ramesh Nallapati, and Christopher D Manning. 2012. Multi-instance multi-label learning for relation extraction. In *Proceedings of the 2012 joint conference on empirical methods in natural language processing and computational natural language learning*, pages 455–465. Association for Computational Linguistics.

Richard S Sutton and Andrew G Barto. 1998. Reinforcement learning: An introduction.

Shingo Takamatsu, Issei Sato, and Hiroshi Nakagawa. 2012. Reducing wrong labels in distant supervision for relation extraction. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pages 721–729. ACL.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008.

Haoyu Wang, Ming Tan, Mo Yu, Shiyu Chang, Dakuo Wang, Kun Xu, Xiaoxiao Guo, and Saloni Potdar. 2019. Extracting multiple-relations in one-pass with pre-trained transformers. *arXiv preprint arXiv:1902.01030*.

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 EMNLP*, pages 1753–1762.

Daojian Zeng, Kang Liu, Siwei Lai, Guangyou Zhou, and Jun Zhao. 2014. Relation classification via convolutional deep neural network. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 2335–2344.

Han Zhang, Ian Goodfellow, Dimitris Metaxas, and Augustus Odena. 2018. Self-attention generative adversarial networks. *arXiv preprint arXiv:1805.08318*.