Information Processing and Management xxx (xxxx) xxxx

Contents lists available at ScienceDirect

Information Processing and Management

journal homepage: www.elsevier.com/locate/infoproman

Deep ranking based cost-sensitive multi-label learning for distant supervision relation extraction

Hai Ye[a], Zhunchen Luo[*,b,c]

[a] School of Computer Science and Engineering, Beihang University, Beijing, China
[b] Information Research Center of Military Science, Beijing, China
[c] PLA Academy of Military Science, Beijing, China

ARTICLE INFO

Keywords:
Distant supervision
Relation extraction
Class ties
Class imbalance
Multi-label learning
Cost-sensitive learning
Deep ranking

ABSTRACT

Knowledge base provides a potential way to improve the intelligence of information retrieval (IR) systems, for that knowledge base has numerous relations between entities which can help the IR systems to conduct inference from one entity to another entity. Relation extraction is one of the fundamental techniques to construct a knowledge base. Distant supervision is a semi-supervised learning method for relation extraction which learns with labeled and unlabeled data. However, this approach suffers the problem of *relation overlapping* in which one entity tuple may have multiple relation facts. We believe that relation types can have latent connections, which we call *class ties*, and can be exploited to enhance relation extraction. However, this property between relation classes has not been fully explored before. In this paper, to exploit class ties between relations to improve relation extraction, we propose a general ranking based multi-label learning framework combined with convolutional neural networks, in which ranking based loss functions with regularization technique are introduced to learn the latent connections between relations. Furthermore, to deal with the problem of *class imbalance* in distant supervision relation extraction, we further adopt cost-sensitive learning to rescale the costs from the positive and negative labels. Extensive experiments on a widely used dataset show the effectiveness of our model to exploit class ties and to relieve class imbalance problem.

1. Introduction

Relation extraction (RE) aims to classify the relations (or called relation facts) between two given named entities from natural-language text. Fig. 1 shows two sentences with the same entity tuple but two different relation facts. RE is to accurately extract the corresponding relation facts (*place_of_birth, place_lived*) for the entity tuple (*Patsy Ramsey, Atlanta*) based on the contexts of sentences. Supervised-learning methods require numerous labeled data to work well. With the rapid growth of volume of relation types, traditional methods can not keep up with the step for the limitation of labeled data. In order to narrow down the gap of data sparsity, Mintz, Bills, Snow, and Jurafsky (2009) proposes *distant supervision (DS)* for relation extraction, which automatically generates training data by aligning a knowledge facts database (ie. Bollacker, Evans, Paritosh, Sturge, & Taylor, 2008) to texts. For a fact (e.g. entity tuple with a relation type) from the knowledge base, the sentences containing the entity tuple in the fact are regarded as the training data.

Class ties mean the connections (relatedness) between relations types for relation extraction. In general, we conclude that class ties

* corresponding author.
E-mail addresses: hye.me@outlook.com (H. Ye), zhunchenluo@gmail.com (Z. Luo).

https://doi.org/10.1016/j.ipm.2019.102096
Received 9 December 2018; Received in revised form 25 July 2019; Accepted 3 August 2019

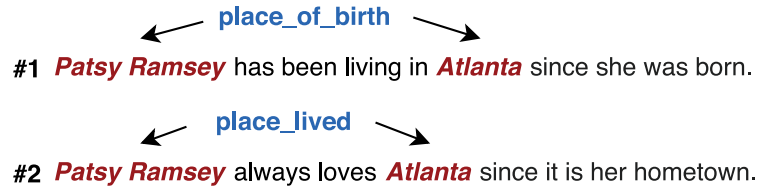

Fig. 1. Training instances generated by freebase. The entity tuple is (*Patsy Ramsey, Atlanta*) and its two relation facts are *palce_of_birth* and *place_lived*.

can have two categories: weak class ties and strong class ties. Weak class ties mainly involve the co-occurrence of relations such as *place_of_birth* and *place_lived, CEO_of* and *founder_of*. Besides, strong class ties mean that relations have latent logical entailments. Take the two relations of *capital_of* and *city_of* for example, if one entity tuple has the relation of *capital_of*, it must express the relation fact of *city_of*, because the two relations have the entailment of *capital_of* $\Rightarrow$ *city_of*. Obviously the opposite induction is not correct. Further take the following sentence of

Jonbenet told me that her mother [Patsy Ramsey]$_{e_1}$ *never left* [Atlanta]$_{e_2}$ *since she was born.*

for example. This sentence expresses two relation facts which are *place_of_birth* and *place_lived*. However, the word "born" is a strong bias to extract *place_of_birth*, so it may not be easy to predict the relation of *place_lived*, but extracting *place_of_birth* will provide evidence for prediction of *place_lived* by incorporating the weak ties between the two relations,

Exploiting class ties is necessary for DS based relation extraction. In DS scenario, there is a challenge that one entity tuple can have multiple relation facts which is called *relation overlapping* (Hoffmann, Zhang, Ling, Zettlemoyer, & Weld, 2011; Surdeanu, Tibshirani, Nallapati, & Manning, 2012), as shown in Fig. 1. However, the relations of one entity tuple can have class ties mentioned above which can be leveraged to enhance relation extraction, for that it narrows down potential searching spaces and reduces uncertainties between relations when predicting unknown relations, such that if one pair of entities has *CEO_of* relation, it will contain *founder_of* relation with high possibility.

To exploit class ties between relations, we propose to make joint extraction by considering *pairwise* connections between positive and negative labels inspired by Fürnkranz, Hüllermeier, Mencía, and Brinker (2008) and Zhang and Zhou (2006). As the example for one entity tuple with two different relation types shown in Fig. 1, by extracting the two relations jointly, we can maintain the *class ties* (co-occurrence) of them and the class ties can be learned by potential models, which can be leveraged to extract instances with unknown relations. We introduce a ranking based multi-label learning framework to make joint extraction, to learn to rank the prediction probability for positive relations higher than negative ones. We design ranking based loss functions for multi-label learning. Furthermore, inspired by Zhou, Zhang, Huang, and Li (2012) and Evgeniou, Micchelli, and Pontil (2005), we add a regularization term to the loss functions to better learn the relatedness between relation facts, and we only regularize the positive relation types ignoring the relation of NR (does not express any relation) based on the assumption that the connections between relations are only in positive relations but not in NR (see Section 3.4).

Besides, class imbalance is the another severe problem which can not be ignored for distant supervision relation extraction. We find that around 70% training data express NR relation type and even more than 90% in test set, so samples with NR type count a much higher proportion comparing to the positive samples (not categorized as NR). This problem will severely affect the model training, causing the model easily to classify the samples to have the NR relation type (Japkowicz & Stephen, 2002). To overcome this problem, based on the ranking loss functions, we further adopt cost-sensitive learning to rescale the costs from the positive and negative labels, by increasing the losses for positive labels and penalizing losses from NR type (detailed in Section 3.5).

Furthermore, combining information across sentences will be more appropriate for joint extraction which provides more information from other sentences to extract each relation (Lin, Shen, Liu, Luan, & Sun, 2016; Zheng, Li, Wang, Yan, & Zhou, 2016). In Fig. 1, sentence #1 is the evidence for *place_of_birth*, but it also expresses the meaning of "living in someplace", so it can be aggregated with sentence #2 to extract *place_lived*. Meanwhile, the word of "hometown" in sentence #2 can provide evidence for *place_of_birth* which should be combined with sentence #1 to extract *place_of_birth*.

In this work, we propose a unified model that integrates ranking based cost-sensitive multi-label learning with convolutional neural network (CNN) to exploit class ties between relations and further relieve the class imbalance problem. Inspired by the effectiveness of deep learning for modeling sentence features (LeCun, Bengio, & Hinton, 2015), we use CNN to encode sentences. Similar to Lin et al. (2016) and Santos, Xiang, and Zhou (2015), we use class embeddings to represent relation classes. The whole model architecture is presented in Fig. 2. We first use CNN to embed sentences, then we introduce two variant methods to combine the embedded sentences into one bag representation vector aiming to aggregate information across sentences, after that we measure the similarity between the bag representation and relation class in real-valued space. Finally, we use the ranking loss functions to learn to make joint extraction over multiple relation types.

Our experimental results on dataset of Riedel, Yao, and McCallum (2010) are evident that: (1) Our model is much more effective than the baselines; (2) Leveraging class ties will enhance relation extraction and our model is efficient to learn class ties by joint extraction; (3) A much better model can be trained after relieving class imbalance from NR.

Our contributions in this paper can be encapsulated as follows:

- We propose to leverage class ties to enhance relation extraction. Combined with CNN, an effective deep ranking based multi-label

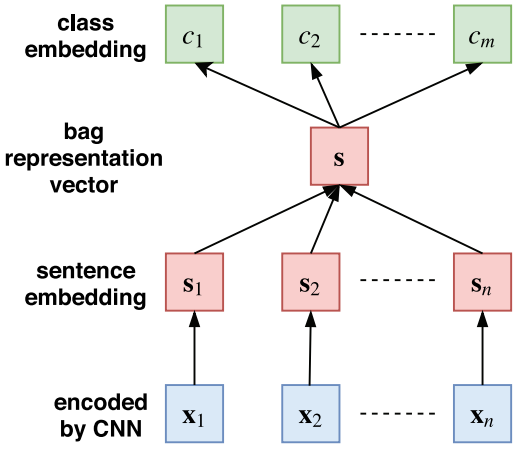

Fig. 2. The main architecture of our model. The features of sentences are encoded by CNN model, and then the sentence embeddings are aggregated, finally the bag representation is used to make joint extraction.

learning model with regularization technique is introduced to exploit class ties.

- We adopt the cost-sensitive learning to relieve the class imbalance problem and experimental results show the effectiveness of our method.

2. Related work

2.1. Relation extraction

Previous methods on relation extraction can mainly be summarized as supervision based and distant supervision based. Supervision based methods needs much labeled data to work well which can not keep up with the rapid growth of relation types. To overcome the problem of data sparsity for supervision based methods, distant supervision relation extraction has been proposed by Mintz et al. (2009). However, DS based relation extraction suffers the two problems of *wrong labelling problem* and *overlapping problem*, in which the former means that sentences containing certain entities actually do not express the relation type of the entities indicated or even do not express any relations and the latter mean that one entity tuple may have multiple relation types. To solve the problem of wrong labelling, Riedel et al. (2010) introduces multi-instance learning for relation extraction in which the mentions of one certain entity tuple are merged as one bag and make the model to extract relations on mention bags, however this method can not deal with the relation overlapping problem. Afterwards, Hoffmann et al. (2011) and Surdeanu et al. (2012) introduce the framework of multi-instance multi-label learning to jointly overcome the two problems and improve the performance significantly. Though they also propose to make joint extraction of relations, they only use information from single sentence losing information from other sentences. Han and Sun (2016) tries to use *Markov logic* model to capture consistency between relation labels, on the contrary, our model leverages deep ranking to learn class ties automatically.

Recent years, deep learning has achieved remarkable success in computer vision and natural language processing (LeCun et al., 2015). Deep learning has been applied to automatically learn the features of sentences (Jiang, Ye, Luo, Chao, & Ma, 2018; Lin et al., 2016; Santos et al., 2015; Ye, Jiang, Luo, & Chao, 2018; Ye & Wang, 2018; Ye, Yan, Luo, & Chao, 2017; Yu Mo & Dredze, 2014; Zeng et al., 2014). In supervision relation extraction, Zeng et al. (2014) applies convolutional neural networks to model sentences and import position feature for RE, which obtains significant gains in RE performance. Afterwards, Yu Mo and Dredze (2014), Santos et al. (2015) and Lin et al. (2016) further introduce more advanced deep learning models for RE. In distant supervision relation extraction, Zeng, Liu, Chen, and Zhao (2015) proposes a piecewise convolutional neural network with multi-instance learning for DS based relation extraction, which improves the precision and recall significantly. Afterwards, Lin et al. (2016) introduces the attention mechanism (Bahdanau, Cho, & Bengio, 2015; Luong, Pham, & Manning, 2015) to merge the sentence features aiming to construct better bag representations. Lin, Liu, and Sun (2017) further proposes a multi-lingual neural relation extraction framework considering the information consistency and complementarity among cross-lingual texts. However, the two deep learning based models only make separated extraction thus can not model class ties between relations. Recently, Zeng, Lin, Liu, and Sun (2016) proposes to incorporate relation paths for distant supervision relation extraction and Ji, Liu, He, and Zhao (2017) introduces to use the description of entities to enhance distant supervision relation extraction. Chen, Feng, Huang, Luo, and Zhao (2018) proposes a joint inference approach by encoding implicit relation requirements for relation extraction. Joint learning is also applied to jointly study two related tasks (Ye, Li, & Wang, 2019). Besides, a lot of works have been proposed in recent times to solve the wrong labelling problem. Luo et al. (2017) proposes to model the noise caused by wrong labelling problem and show that dynamic transition matrix can effectively characterize the noises. Qin, Xu, and Wang (2018a) and Han, Liu, and Sun (2018) propose to use adversarial learning (Goodfellow et al., 2014) to solve the wrong labelling problem. Instead, Feng, Huang, Zhao, Yang, and Zhu (2018), Qin, Xu, and Wang (2018b) adopt reinforcement learning to learn to select high-quality data for training. Liu, Wang,

Chang, and Sui (2017) dynamically corrects the wrong labeled data during training by exploiting semantic information from labeled entity pairs. Liu, Zhang, Zhou, and Jia (2018) transfer the priori knowledge learned from relevant entity classification task to make the model robust to noisy data.

2.2. Deep learning to rank

Learning to rank (LTR) is an important technique in information retrieval (IR) (Liu, 2009). The methods to train a LTR model include pointwise, pairwise and listwise. We apply pairwise LTR in our paper. Deep learning to rank has been widely used in many problems to serve as a classification model. In image retrieval, Zhao, Huang, Wang, and Tan (2015) applies deep semantic ranking for multi-label image retrieval. In text matching, Severyn and Moschitti (2015) adopts learning to rank combined with deep CNN for short text pairs matching. In traditional supervised relation extraction, Santos et al. (2015) designs a pairwise loss function based on CNN for single label relation extraction. Based on the advantage of deep learning to rank, we propose pairwise learning to rank (LTR) (Liu, 2009) combined with CNN in our model aiming to jointly extract multiple relations.

2.3. Cost-sensitive learning

Cost-sensitive learning is one of the techniques for class imbalance problem, which assigns higher wrong classification costs to classes with small proportion. For example, Shen, Wang, Wang, Bai, and Zhang (2015) proposes a regularized softmax to deal with the imbalanced edge label classification. Khan, Bennamoun, Sohel, and Togneri (2015) adopts cost-sensitive learning to learn deep feature representations from imbalanced data. Another approach to relieve class imbalance problem is re-sampling (He & Garcia, 2009; Huang, Li, Change Loy, & Tang, 2016) including over-sampling and under-sampling, which aims to balance the distributions of data in different labels.

This paper is the extension of Ye, Chao, Luo, and Li (2017). Compared to original work in Ye, Chao, et al. (2017), this paper has several improvements:

Methods: (a) We further fully consider the class imbalance problem. We propose a novel ranking based cost-sensitive loss function combined with multi-label learning. (b) To better learn class ties between relations, we further introduce a regularization term to ranking loss functions.

Experiments: (a) We further do experiments to analyze the effectiveness of our novel cost-sensitive ranking loss functions. (b) The evaluation experiments on the effectiveness of regularization have further be conducted.

Content: (a) We rewrite the description of our methods from the view of multi-label learning and cost-sensitive learning to gain more theoretical justification improvement.

3. Methodology

We introduce our methods in this section. Firstly, we describe the widely used CNN architecture for sentence encoding. Then we discuss the ranking based multi-label learning framework with regularization technique. After that, we introduce the proposed cost-sensitive learning to overcome the NR effects for model training.

3.1. Notation

We define the relation classes as $\mathcal{L} = \{1, 2, \cdots, C\}$, entity tuples as $\mathcal{T} = \{t_i\}_{i=1}^{M}$ and mentions[1] as $\mathcal{X} = \{x_i\}_{i=1}^{N}$. Dataset is constructed as follows: for entity tuple $t_i \in \mathcal{T}$ and its relation class set $L_i \subseteq \mathcal{L}$, we collect all the mentions X_i that contain t_i, the dataset we use is $\mathcal{D} = \{(t_i, L_i, X_i)\}_{i=1}^{H}$. Given a data $(t_k, L_k, X_k) \in \mathcal{D}$, the sentence embeddings of X_k encoded by CNN are defined as $S_k = \{s_i\}_{i=1}^{|X_k|}$ and we use class embeddings $W \in R^{|\mathcal{L}| \times d}$ to represent the relation classes, which will be learned in model training.

3.2. CNN for sentence embedding

We take the effective piecewise CNN architecture adopted from Zeng et al. (2015) and Lin et al. (2016) to encode sentence and we will briefly introduce PCNN in this section. More details of PCNN can be obtained from previous work.

3.2.1. Words representations

- **Word Embedding** Given a word embedding matrix $V \in \mathbb{R}^{l^w \times d^1}$ where l^w is the size of word dictionary and d^1 is the dimension of word embedding, the words of a mention $x = \{w_1, w_2, \cdots, w_n\}$ will be represented by real-valued vectors from V.
- **Position Embedding** The position embedding of a word measures the distance from the word to entities in a mention. We add position embeddings into words representations by appending position embedding to word embedding for every word. Given a position embedding matrix $P \in \mathbb{R}^{l^p \times d^2}$ where l^p is the number of distances and d^2 is the dimension of position embeddings, the dimension of words representations becomes $d^w = d^1 + d^2 \times 2$.

[1] The sentence containing one certain entity is called mention.

3.2.2. Convolution, piecewise max-pooling

After transforming words in x to real-valued vectors, we get the sentence $q \in \mathbb{R}^{n \times d^w}$. The set of kernels K is $\{K_i\}_{i=1}^{d^s}$ where d^s is the number of kernels. Define the window size as d^{win} and given one kernel $K_k \in \mathbb{R}^{d^{win} \times d^w}$, the convolution operation is defined as follows:

$$m_{[i]} = q_{[i:i+d^{win}-1]} \odot K_k + b_{[k]} \tag{1}$$

where m is the vector after conducting convolution along q for $n - d^{win} + 1$ times and $b \in \mathbb{R}^{d^s}$ is the bias vector. For these vectors whose indexes out of range of $[1, n]$, we replace them with zero vectors.

By piecewise max-pooling, when pooling, the sentence is divided into three parts: $m_{[p_0:p_1]}$, $m_{[p_1:p_2]}$ and $m_{[p_2:p_3]}$ (p_1 and p_2 are the positions of entities, p_0 is the beginning of sentence and p_3 is the end of sentence). This piecewise max-pooling is defined as follows:

$$z_{[j]} = max\left(m_{[p_{j-1}:p_j]}\right) \tag{2}$$

where $z \in \mathbb{R}^3$ is the result of mention x processed by kernel K_k; $1 \le j \le 3$. Given the set of kernels K, following the above steps, the mention x can be embedded to o where $o \in R^{d^s * 3}$.

3.2.3. Non-linear layer, regularization

To learn high-level features of mentions, we apply a non-linear layer after pooling layer. After that, a dropout layer is applied to prevent over-fitting. We define the final fixed sentence representation as $r \in \mathbb{R}^{d^f}$ ($d^f = d^s * 3$).

$$s = g(o) \circ h \tag{3}$$

where $g(\cdot)$ is a non-linear function and we use $tanh(\cdot)$ in this paper; h is a Bernoulli random vector with probability p to be 1.

3.3. Combine information across sentences

We propose two options to combine sentences to provide enough information for multi-label learning.

- **AVE** The first option is average method. This method regards all the sentences equally and directly average the values in all dimensions of sentence embedding. This **AVE** function is defined as follows:

$$r = \frac{1}{n} \sum_{s_i \in S_k} s_i \tag{4}$$

  where n is the number of sentences and r is the bag representation combining all sentence embeddings. Because it weights the importance of sentences equally, this method may bring much noise data from two aspects: (1) the wrong labelling data; (2) irrelated mentions for one relation class, for all sentences containing the same entity tuple being combined together to construct the bag representation.
- **ATT** The second one is a sentence-level attention algorithm used by Lin et al. (2016) to measure the importance of sentences aiming to relieve the wrong labelling problem. For every sentence, **ATT** will calculate a weight by comparing the sentence to one relation. We first calculate the similarity between one sentence embedding and relation class as follows:

$$e_j = a \cdot W_{[c]} \cdot s_j \tag{5}$$

  where e_j is the similarity between sentence embedding s_j and relation class c and a is a bias factor. In this paper, we set a as 0.5. Then we apply Softmax to rescale $\mathbf{e}$ ($\mathbf{e} = \{e_i\}_{i=1}^{|X_k|}$) to [0,1]. We get the weight α_j for s_j as follows:

$$\alpha_j = \frac{\exp(e_j)}{\sum_{e_i \in \mathbf{e}} \exp(e_i)} \tag{6}$$

  so the function to merge r with **ATT** is as follows:

$$r = \sum_{i=1}^{|X_k|} \alpha_i \cdot s_i \tag{7}$$

3.4. Learning class ties via ranking based multi-label learning with regularization

Firstly, we have to present the score function to measure the similarity between bag representation r and relation c.

- **Score Function** We use dot function to produce score for r to be predicted as relation c. The score function is as follows:

$$\mathcal{F}(r, c) = W_{[c]} \cdot r \tag{8}$$

  There are other options for score function. In Wang, Cao, de Melo, and Liu (2016), they propose a margin based loss function that measures the similarity between r and $W_{[c]}$ by distance. Because score function is not an important issue in our model, we adopt

dot function, also used by Santos et al. (2015) and Lin et al. (2016), as our score function.Now we start to introduce the ranking loss functions.Pairwise ranking aims to learn the score function $\mathcal{F}(r, c)$ that ranks positive classes higher than negative ones. This goal can be summarized as follows:

$$\forall\, c^+ \in L_k, \forall\, c^- \in \mathcal{L} - L_k : \mathcal{F}(r, c^+) > \mathcal{F}(r, c^-) + \beta \tag{9}$$

where β is a margin factor which controls the minimum margin between the positive scores and negative scores. Inspired by Santos et al. (2015), given c^+ and c^-, we adopt the following function to learn the score function:

$$\begin{aligned}\mathcal{H}(c^+, c^-, r) = &\ln(1 + \exp(\rho[0, \sigma^+ - \mathcal{F}(r, c^+)])) \\ &+ \ln(1 + \exp(\rho[0, \sigma^- + \mathcal{F}(r, c^-)]))\end{aligned} \tag{10}$$

where $[0, \cdot] = \max(0, \cdot)$, ρ is the rescale factor, σ^+ is positive margin and σ^- is negative margin. This loss function is designed to rank positive classes higher than negative ones controlled by the margin of $\sigma^+ - \sigma^-$. In reality, $\mathcal{F}(r, c^+)$ will be higher than σ^+ and $\mathcal{F}(r, c^-)$ will be lower than σ^-. In our work, we set ρ as 2, σ^+ as 2.5 and σ^- as 0.5 adopted from Santos et al. (2015). To simplify the loss functions given in the followings, we use $\rho[0, \sigma^+ - \mathcal{F}(r, c^+)]$ to replace the first term in $\mathcal{H}$ and use $\rho[0, \sigma^- + \mathcal{F}(r, c^-)]$ to replace the second term.To model the class ties (co-occurrence) of the labels, we have the assumption that the positive labels have the same class ties and are connected with each other. Out of this assumption, we have two mechanisms to learn the class ties, which are making joint extraction of relations and explicitly modeling the connections by regularizing the learning of positive labels. In the followings, we will first introduce the loss functions for multi-label learning extended from Eq. (10); then we discuss the regularization term.To learn class ties between relations, we firstly extend the Eq. (10) to make multi-label learning. Followings are the proposed ranking based loss functions:

- **with AVE (Variant-1)** We define the margin-based loss function with option of **AVE** to aggregate sentences as follows:

$$\begin{aligned}G_{[\text{ave}]} = &\sum_{c^+ \in L_k} \rho[0, \sigma^+ - \mathcal{F}(r, c^+)] \\ &+ \rho|L_k|[0, \sigma^- + \mathcal{F}(r, c^-)]\end{aligned} \tag{11}$$

Similar to Weston, Bengio, and Usunier (2011) and Santos et al. (2015), we update one negative class at every training round but to balance the loss between positive classes and negative ones, we multiply $|L_k|$ before the right term in Eq. (11) to expand the negative loss. We apply mini-batch based stochastic gradient descent (SGD) to minimize the loss function. The negative class is chosen as the one with highest score among all negative classes (Santos et al., 2015), i.e.:

$$c^- = \underset{c \in \mathcal{L} - L_k}{\text{argmax}} \mathcal{F}(r, c) \tag{12}$$

- **with ATT (Variant-2)** Now we define the loss function for the option of **ATT** to combine sentences as follows:

$$\begin{aligned}G_{[\text{att}]} = &\sum_{c^+ \in L_k} \Big\{ \rho[0, \sigma^+ - \mathcal{F}(r^{c^+}, c^+)] \\ &+ \rho[0, \sigma^- + \mathcal{F}(r^{c^+}, c^-)]\Big\}\end{aligned} \tag{13}$$

where r^c means the attention weighted representation r where attention weights are merged by comparing sentence embeddings with relation class c and c^- is chosen by the following function:

$$c^- = \underset{c \in \mathcal{L} - L_k}{\text{argmax}} \mathcal{F}(r^{c^+}, c) \tag{14}$$

which means we update one negative class in every training round. We keep the values of ρ, σ^+ and σ^- same as values in Eq. (11). In Eq. (13), for every $c^+ \in L_k$, we need to sample $c^- \in \mathcal{L} - L_k$ according to Eq. (14), so different from Eq. (11), we do not extend the negative loss by multiplying $|L_k|$.

According to this loss function, we can see that: for each class $c^+ \in L_k$, it will capture the most related information from sentences to merge r^{c^+}, then rank $\mathcal{F}(r^{c^+}, c^+)$ higher than all negative scores which each is $\mathcal{F}(r^{c^+}, c^-)$ $(c^- \in \mathcal{L} - L_k)$. We use the same update algorithm to minimize this loss.

Based on the assumption that all positive labels have the same class ties, making joint extraction of the relations can capture the co-occurrence of the labels. If the relations for the same entity pair usually appear together, then extracting them jointly can learn the statistical property of their co-appearance.

- **Regularization** To learn the class ties between relations, we have proposed the ranking based loss functions above. Inspired by Zhou et al. (2012) and Evgeniou et al. (2005), we further capture the relation connections by adding an extra regularization term to the loss functions. We only consider the relatedness between positive labels ignoring NR. The relatedness is measured by the mean function W_{ave}:

$$W_{ave} = \frac{1}{T} \sum_{c \in \mathcal{L} - c_{\text{NR}}} W_{[c]} \tag{15}$$

where $T = |\mathcal{L} - c_{\text{NR}}|$. W_{ave} is the center of the labels, and we hope the positive labels can be close to the center which can be

measured by:

$$\frac{1}{T}\sum_{c\in\mathcal{L}-c_{NR}}\|W_{[c]}-W_{ave}\|_2 \tag{16}$$

Following Zhou et al. (2012), to model the class ties we need to minimize the loss function as follows:

$$\Theta(W)=\epsilon\|W_{ave}\|_2+\eta\frac{1}{T}\sum_{\mathcal{L}-c_{NR}}\|W_{[c]}-W_{ave}\|_2 \tag{17}$$

where ϵ and η are hyper-parameters. Eq. (17) is designed based on the consideration that the labels in which class ties exist should be clustered together and should be close to the center of these labels. According to Eq. (15), Eq. (16) can be re-written as:

$$-\|W_{ave}\|_2+\frac{1}{T}\sum_{c\in\mathcal{L}-c_{NR}}\|W_{[c]}\|_2 \tag{18}$$

By merging Func. (18) into Eq. (17), we have the our final regularization term:

$$\Theta(W)=\epsilon\|W_{ave}\|_2+\eta\frac{1}{T}\sum_{c\in\mathcal{L}-c_{\mathrm{NR}}}\|W_{[c]}\|_2 \tag{19}$$

In this paper, we set η as 10^{-3} and ϵ is set as 10^{-6}.

3.5. Ranking based cost-sensitive multi-label learning

In relation extraction, the dataset will always contain certain negative samples which do not express any relation types and are classified as NR type (no relation). Table 1 presents the proportion of NR samples in the dataset from Riedel et al. (2010), which shows that the almost data is about NR. Data imbalance will severely affect the model training and cause the model only sensitive to classes with high proportion (He & Garcia, 2009), causing a positive sample to be classified as NR. In order to relieve this problem, we adopt cost-sensitive learning to construct the loss function. Based on $G_{[\mathrm{att}]}$, the cost-sensitive loss function which is **Variant-3** is as follows:

$$\begin{aligned}G_{[\mathrm{cost\_att}]}=\sum_{c^*\in L_k}\Bigg\{&g(c^*)\Big(\rho[0,\sigma^+-\mathcal{F}(r^{c^*},c^*)]\Big)\\&+\rho[0,\sigma^-+\mathcal{F}(r^{c^*},c^-)]\\&+\sum_{c^+\in L_k-c^*}\gamma\rho[0,\sigma^+-\mathcal{F}(r^{c^*},c^+)]\\&+\gamma\mathbf{1}(c^*\neq c_{\mathrm{NR}})\rho[0,\sigma^-+\mathcal{F}(r^{c^*},c_{\mathrm{NR}})]\Bigg\}\end{aligned} \tag{20}$$

where $g(c^*)=\mathbf{1}(c=c_{\mathrm{NR}})\lambda+\mathbf{1}(c\neq c_{\mathrm{NR}})1$; $\mathbf{1}(\cdot)$ is an indicate function. Similar to Eq. (14), we select c^- as follows:

$$c^-=\underset{c\in\mathcal{L}-L_k}{\mathrm{argmax}}\mathcal{F}(r^{c^*},c) \tag{21}$$

Because NR counts a high proportion in the training set, without controlling, the model will receive large costs from NR. In order to relieve the effects from NR, we penalize the losses from NR. Specifically, we have two strategies to do that. We adopt two hyper-parameters which are λ ($\lambda<1$) and γ to penalize the losses from NR. If $c^*\in L_k$ is a positive label, to balance the costs between the positive labels and the NR label, we further add the costs from the left positive relations $c^+\in L_k-c^*$ and at the same time, the extra cost from NR is calculated. The default value of γ is 1 and if γ is small enough, this loss function will be similar to loss Eq. (13). Based on the experimental results, we find that the best results are achieved when λ is set to 0, so we set λ as 0 in this paper. How the λ and γ affect model performance is discussed in Sections 4.5 and 4.6. We also add the regularization term $\Theta(W)$ to $G_{[\mathrm{cost\_att}]}$ to better capture the class ties between relations.

We give out the pseudocode of merging $G_{[\mathrm{cost\_att}]}$ in Algorithm 1.

4. Experiments

In this section, we conduct two sets of experiments, in which the first one is for comparing our method with the baselines and the second one is used to evaluate our model. Without the special statement, we will adhere to the methods and settings mentioned above

Table 1
The proportions of NR samples from Riedel's dataset.

Pro. (%)	Training	Test
Riedel	72.52	96.26

input : $\mathcal{L}$, (t_k, L_k, X_k) and S_k;
output: $G_{[\text{cost\_att}]}$;
$G_{[\text{cost\_att}]} \leftarrow 0$;
for $c^* \in L_k$ **do**
  Merge representation r^{c^*} by Eq. 5, 6, 7;
  $G_{[\text{cost\_att}]} \leftarrow g(c^*)(\rho[0, \sigma^+ - \mathcal{F}(r^{c^*}, c^*)])$;
  $c^- \leftarrow \arg\max_{c \in \mathcal{L} - L_k} \mathcal{F}(r^{c^*}, c)$;
  $G_{[\text{cost\_att}]} \leftarrow G_{[\text{cost\_att}]} + \rho[0, \sigma^- + \mathcal{F}(r^{c^*}, c^-)]$;
  for $c^+ \in L_k - c^*$ **do**
    $G_{[\text{cost\_att}]} \leftarrow G_{[\text{cost\_att}]} + \gamma\rho[0, \sigma^+ - \mathcal{F}(r^{c^*}, c^+)]$;
  $G_{[\text{cost\_att}]} \leftarrow G_{[\text{cost\_att}]} + \gamma \mathbf{1}(c^* \neq c_{\text{NR}})\rho[0, \sigma^- + \mathcal{F}(r^{c^*}, c_{\text{NR}})]$;
return $G_{[\text{cost\_att}]}$;

Algorithm 1. Ranking based Cost-sensitive Multi-label Learning.

to conduct the following experiments.

4.1. Dataset and evaluation criteria

4.1.1. Dataset

We conduct our experiments on a widely used dataset, developed by Riedel et al. (2010) and has been used by Hoffmann et al. (2011), Surdeanu et al. (2012), Zeng et al. (2015) and Lin et al. (2016). The dataset aligns Freebase relation facts with the New York Times corpus, in which training mentions are from 2005 to 2006 corpus and test mentions from 2007. The training set contains 522,611 sentences, 281,270 entity pairs and 18,252 relation facts. In test set, there are 172,448 sentences, 96,678 entity pairs and 1950 relation facts. In all, there are 53 relation labels including the NR relation. Following Mintz et al. (2009), we adopt held-out evaluation framework in all experiments. We use all training dataset to train our model and then test the trained model on test dataset to compare the predicted relations to gold relations.

4.1.2. Evaluation criteria

To evaluate the model performance, we draw the precision/recall (P/R) curves and precision@N (P@N) is reported to illustrate the model performance. For the metric of P/R curve, the bigger of the area contained under the curve, the better of the model performance.

4.2. Experimental settings

4.2.1. Word embeddings

We adopt the trained word embeddings from Lin et al. (2016). Similar to Lin et al. (2016), we keep the words that appear more than 100 times to construct word dictionary and use “UNK” to represent the other ones.

4.2.2. Hyper-parameter settings

Three-fold validation on the training dataset is adopted to tune the parameters following Surdeanu et al. (2012). We select word embedding size from {50, 100, 150, 200, 250, 300}. Batch size is tuned from {80, 160, 320, 640}. We determine learning rate among {0.01, 0.02, 0.03, 0.04}. The window size of convolution is tuned from {1, 3, 5}. We keep other hyper-parameters same as Zeng et al. (2015): the number of kernels is 230, position embedding size is 5 and dropout rate is 0.5. Table 2 shows the detailed parameter settings.

4.3. Comparisons with baselines

4.3.1. Baseline

We compare our model with the following baselines:

- **Mintz** (Mintz et al., 2009) is the first original model which incorporates distant supervision for relation extraction.
- **MultiR** (Hoffmann et al., 2011) is the multi-instance learning based graphical model which aims to address overlapping relation problem.
- **MIML** (Surdeanu et al., 2012) is a multi-instance multi-label framework which jointly considers the wrong labelling problem and overlapping problem.
- **PCNN + ATT** (Lin et al., 2016) is the previous state-of-the-art model in dataset of Riedel et al. (2010) which applies sentence-level attention to relieve the wrong labelling problem in DS based relation extraction. This model applies piece-wise convolutional neural network (Zeng et al., 2015) to model sentences.

Besides comparing to the above methods, we also compare our variant models represented by **Rank + AVE** (using loss function of $G_{[ave]}$), **Rank + ATT** (using loss of $G_{[att]}$)and **Rank + Cost** (using loss of $G_{[cost_att]}$).

Table 2
Hyper-parameter settings.

Parameter name	Symbol	Value
Window size	d^{win}	3
Sentence. emb. dim.	d^{f}	690
Word. emb. dim.	d^{1}	50
Position. emb. dim.	d^{2}	5
Batch size	$\mathcal{B}$	160
Learning rate	μ	0.03
Dropout pos.	p	0.5

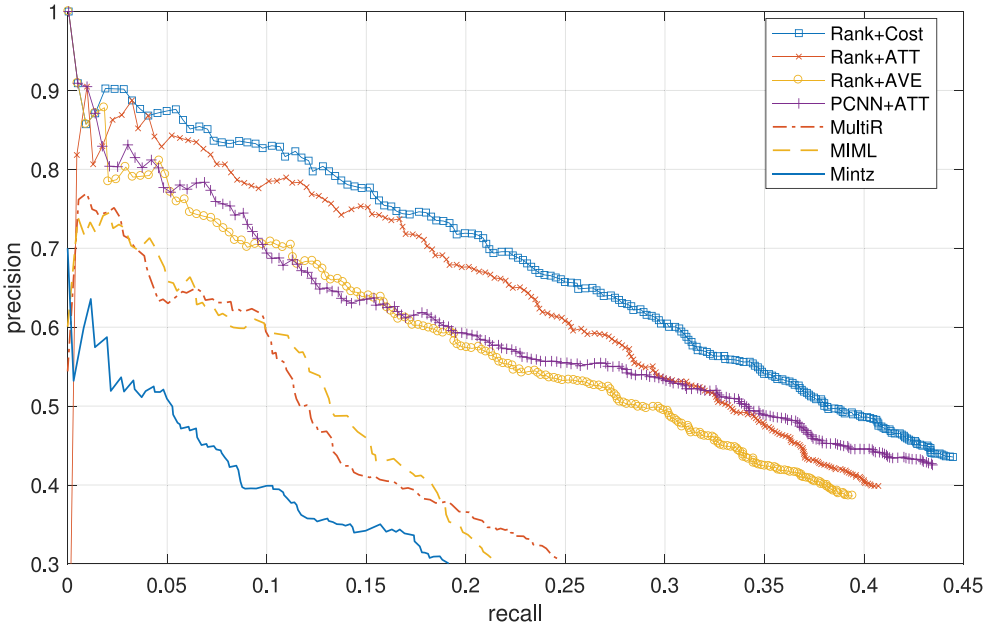

Fig. 3. Performance comparison of our model and the baselines. "Rank + Cost" is using the loss function of $G_{[\text{cost\_att}]}$, "Rank + ATT" is using $G_{[\text{att}]}$ and "Rank + AVE" is using $G_{[\text{ave}]}$.

4.3.2. Results and discussion

We compare our three variants of loss functions with the baselines and the results are shown in Fig. 3. From the results we can see that:

- Rank + AVE (Variant-1) lags behind PCNN + ATT, whose reason may lie in that Rank + AVE does not use the attention mechanism to aggregate the information among the sentences, which brings much noise for encoding sentence contexts;
- After adopting the attention mechanism, Rank + ATT achieves much better performances comparing to Rank + AVE, and even better than PCNN + ATT;
- Comparing PCNN + ATT and Rank + ATT, we can see that Rank + ATT is superior to PCNN + ATT, which comes from the strategy that we model the class ties into the relation extraction;
- Our variant method of Rank + Cost achieves the best performance among all the baselines; by comparing to Rank + ATT, our cost-sensitive learning method can really work for relieving the negative effects from NR.

4.4. Impact of class ties

In this section, we conduct experiments to reveal the effectiveness of our model to learn class ties with three variant loss functions mentioned above, and the impact of class ties for relation extraction. As mentioned above, we adopt two techniques to model the class ties: multi-label learning with ranking based loss functions and regularization term to better model class ties. In the followings, we will conduct experiments to reveal the two aspects for modeling class ties. We will adopt P/R curves and precisions@N (100, 200, ⋯, 500) to show the model performances.

- **Ranking based loss function.** The effectiveness of ranking loss functions to learn class ties lies in the joint extraction of relations to conduct multi-label leaning, so to reveal the impact of ranking loss function to learn class ties, we will compare the joint extraction with separated extraction. Regularization term is added to all variant models. To conduct the experiment of separated extraction, we divide the labels of entity tuple into single label and for one relation label we select the sentences expressing this relation to construct the bag, then we use the re-constructed dataset to train our model with our three variant loss functions. Experimental results are shown in Fig. 4 and Table 3. From the results we can see that: (1) For Rank + ATT and Rank + Cost, joint extraction exhibits better performance than separated extraction, which demonstrates class ties will improve relation extraction and the two methods are effective to learn class ties; (2) For Rank + AVE, surprisingly joint extraction does not keep up with separated extraction. For the second phenomenon, it may come from the strategy of **AVE** method to aggregate sentences. To make joint extraction, we will combine all the sentences containing the same entity tuple, however, not all sentences have the same relation, the fact is that one part of the sentences express one relation type and some will have another one. Simply averaging the sentence representations will hinder the model to learn the latent mapping from the sentences to the corresponding relation type, because averaging operation will gender redundant information from other unrelated sentences.
- **Regularization.** To see the impact of regularization technique for modeling class ties, we compare the methods using

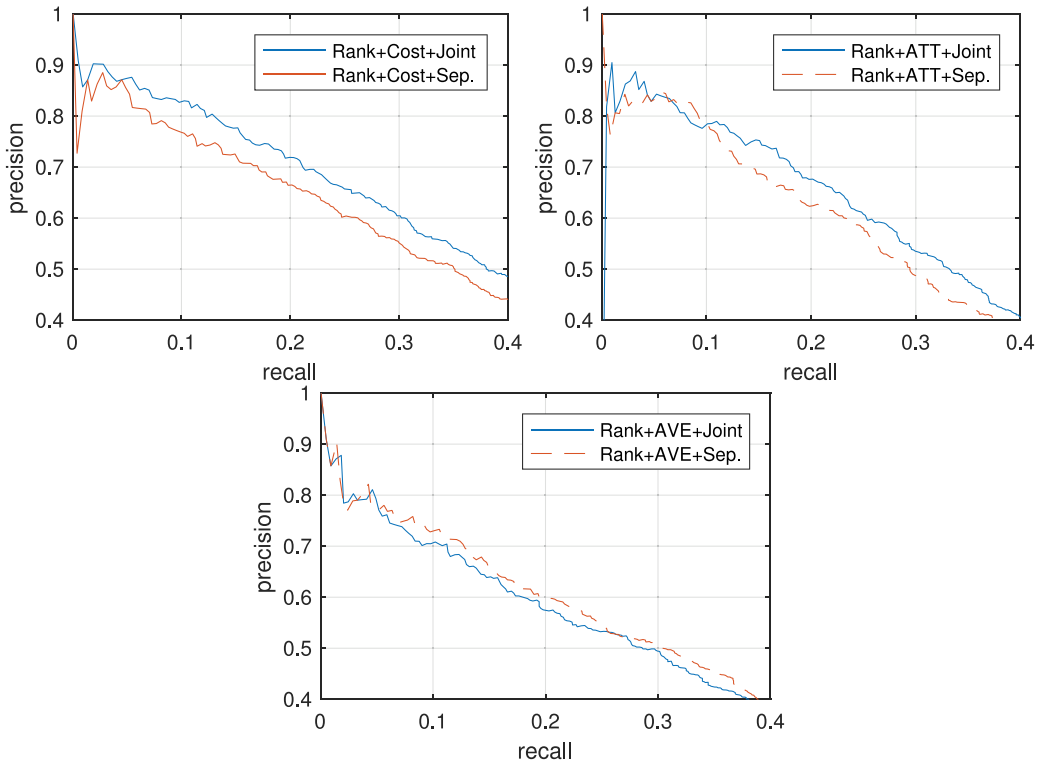

Fig. 4. Results for impact of ranking based loss function with methods of Rank + AVE, Rank + ATT and Rank + Cost.

Table 3
Precisions for top 100, 200, 300, 400, 500 and average of them for impact of joint extraction and class ties.

P@N(%)	100	200	300	400	500	Ave.
R. + AVE + J.	79.1	73.8	70.4	66.0	63.1	70.5
R. + AVE + S.	**80.2**	**74.9**	**72.2**	**67.8**	**64.0**	**71.8**
R. + ATT + J.	**86.8**	80.6	**78.4**	**75.2**	**71.1**	**78.4**
R. + ATT + S.	82.4	**82.7**	75.3	70.1	66.2	75.3
R. + ExATT + J.	**86.8**	**83.2**	**81.1**	**76.7**	**73.5**	**80.3**
R. + ExATT + S.	85.7	78.5	75.6	72.4	69.0	76.3

regularization with the ones without using regularization. All variant models are in setting of joint extraction. The results are shown in Fig. 5 and Table 4. From the results, we can see that after regularizing the learning of relations, the model performance can be further improved indicated by methods of Rank + Cost and Rank + ATT, which demonstrates the effectiveness of regularization to model class ties. We do not see many effects of regularization for method of Rank + AVE. Noises brought by averaging sentence embeddings may hinder the positive effects of regularization.

4.5. Impact of cost-sensitive learning

In this section, we conduct experiments to reveal the effectiveness of cost-sensitive learning to relieve the impact of NR for model training and model performance. For the loss function of $G_{[\text{cost\_att}]}$, we have two parts for cost-sensitive learning: the first is the one penalized by γ, and the second is the NR cost penalized by λ. Based on the loss function of Variant-3, we respectively relieve the cost controlled by γ and the cost of NR controlled by λ to see the impact of cost-sensitive learning. We will adopt P/R curves and precisions@N (100, 200, ⋯, 500) to show the model performances.

The results are shown in Fig. 6 and Table 5. From the results, we can see that considering the cost controlled by γ can sightly improve the performance in low recall range and considering the cost of NR controlled by λ can boost the performance significantly. Considering both of the two kinds of costs can achieve the best performance. From these results, we can see that relieving NR impact is really important to improve the extraction performance.

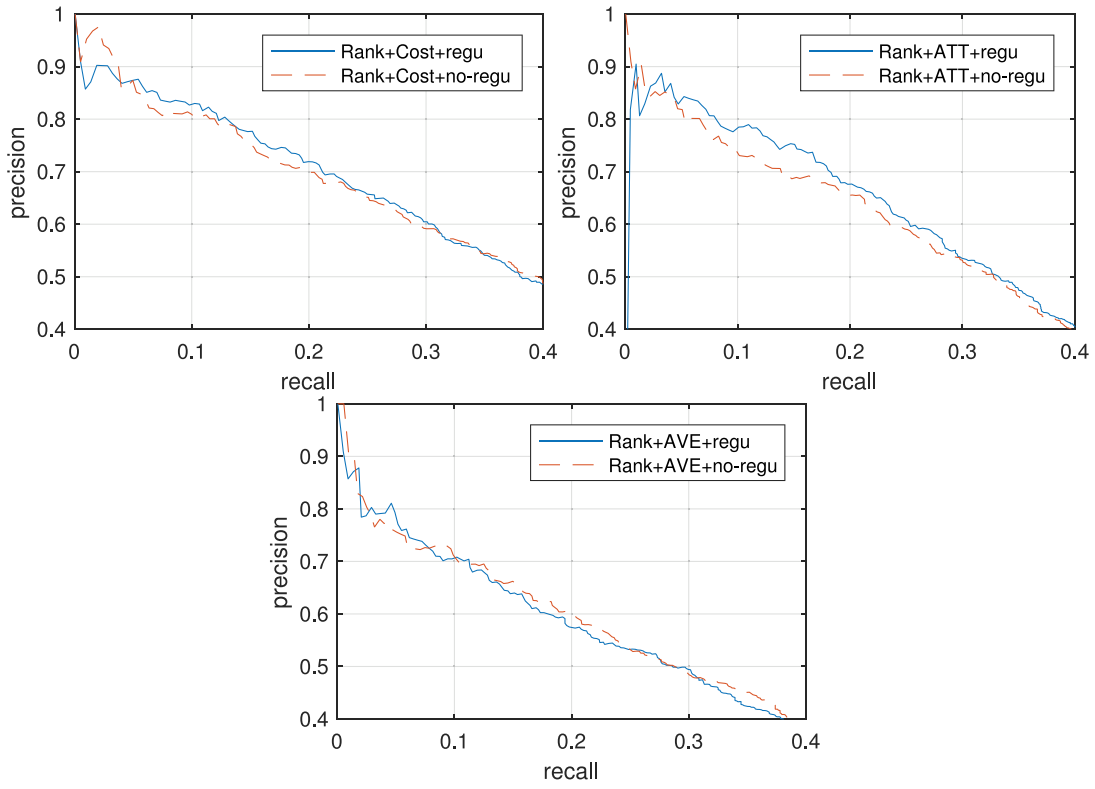

Fig. 5. Results for impact of regularization to model class ties.

Table 4
Precisions for top 100, 200, 300, 400, 500 and average of them for impact of regularization to model class ties.

P@N(%)	100	200	300	400	500	Ave.
R. + AVE + no-regu.	78.0	72.3	69.8	**66.5**	**64.0**	70.1
R. + AVE + regu.	**79.1**	**73.8**	**70.4**	66.0	63.1	**70.5**
R. + ATT + no-regu.	84.6	77.5	72.9	69.6	68.0	74.5
R. + ATT + regu.	**86.8**	**80.6**	**78.4**	**75.2**	**71.1**	**78.4**
R. + Cost + no-regu.	85.7	81.7	80.1	75.2	71.3	78.8
R. + Cost + regu.	**86.8**	**83.2**	**81.1**	**76.7**	**73.5**	**80.3**

4.6. Impact of NR

From the discussion above, we can know that NR can have much significant impact for model performance, so in this section, we conduct more experiments to reveal the impact of NR cost controlled by λ for model performance.

- **Effect of λ penalty.** We conduct experiments on the choice of λ. Based on the loss function of Variant-3, we select λ from {0, 0.001, 0.01, 0.1} to see how much effect of NR can gender to the performance. We also adopt P/R curves and precisions@N (100, 200, ⋯, 500) to show the model performances. Models are set with joint extraction and regularization. The results are shown in Fig. 7 and Table 6. From the results we can find that when λ becomes larger (from 0 to 0.1), the model performance will decrease because NR will have more negative impact on model performance, so in order to achieve better model performance, the value of λ should be set smaller.
- **Effect of NR for model convergence.**Then we further evaluate the impact of NR for convergence behavior of our model in model training. Also with the three variant loss functions, in each iteration, we record the maximal value of F-measure[2]to represent the model performance at current epoch. Models are with setting of joint extraction but without regularization. Model parameters are tuned for 15 times and the convergence curves are shown in Fig. 8. From the result, we can find out: "+NR" converges quicker than "-NR" and arrives to the final score at the around 11 or 12 epoch. In general, "−NR" converges more smoothly and will achieve better performance than "+NR" in the end.

[2] $F = 2*P*R/(P + R)$.

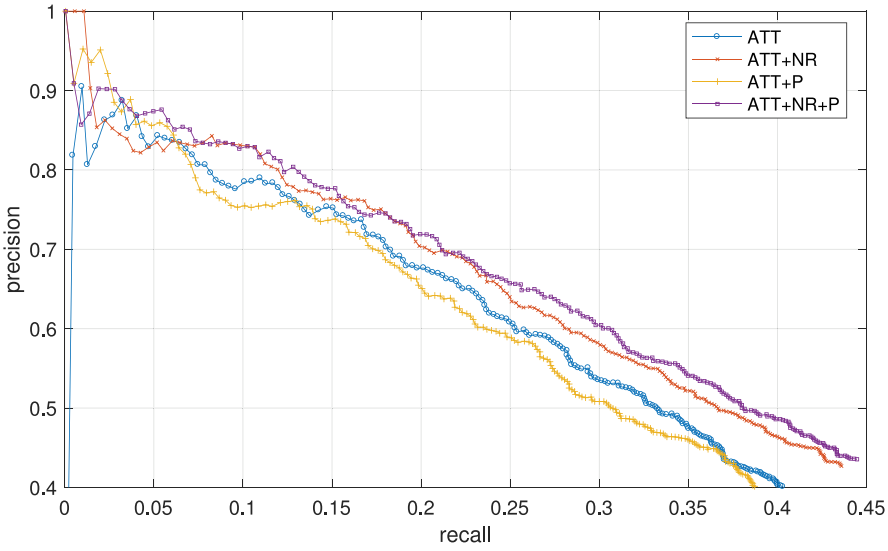

Fig. 6. Results for impact of cost-sensitive learning. "ATT" means the loss function of Variant-2; "ATT + NR" means only considering the cost of NR controlled by λ ignoring the cost controlled by γ based on Variant-2 and λ is set to 0; "ATT + P" means considering the cost controlled by γ based on Variant-2 ignoring the cost of NR and γ is set to 1; "ATT + NR + P" is the loss function of Variant-3 and jointly considers the two kinds of costs mentioned above, λ is set to 0 and γ is 1.

Table 5
Precisions for top 100, 200, 300, 400, 500 and average of them for impact of cost-sensitive learning.

P@N(%)	100	200	300	400	500	Ave.
ATT	86.8	80.6	78.4	75.2	71.1	78.4
ATT + NR	82.4	84.3	80.1	76.2	73.5	79.3
ATT + P	85.7	77.5	75.6	73.7	69.9	76.5
ATT + NR + P	86.8	83.2	81.1	76.7	73.5	80.3

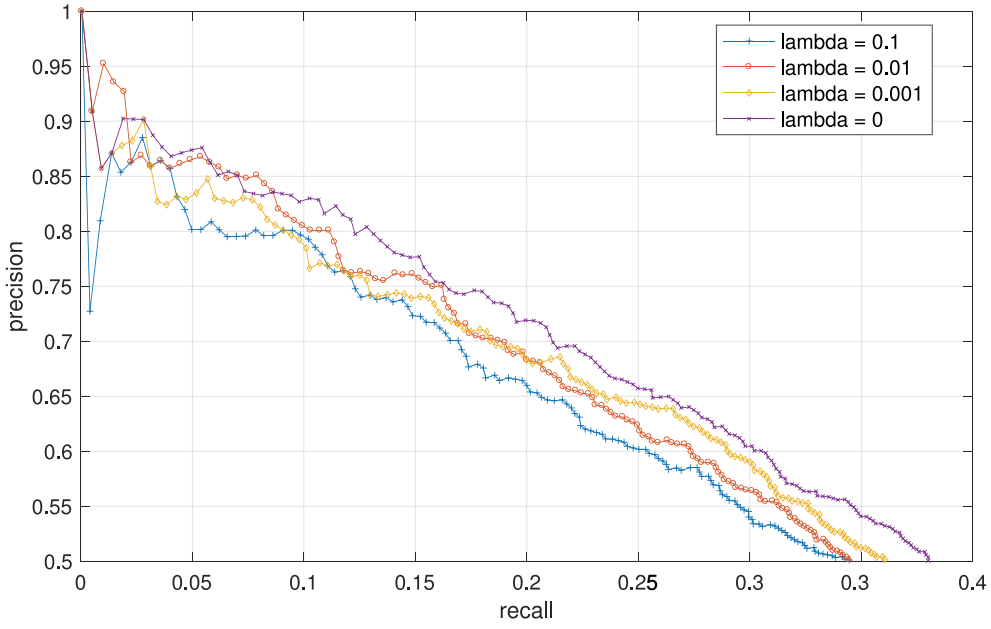

Fig. 7. Effect of λ for model performance based on the loss function of Variant-3.

Table 6
Precisions for top 100, 200, 300, 400, 500 and average of them for impact of cost-sensitive learning.

P@N(%)	100	200	300	400	500	Ave.
λ = 0	86.8	83.2	81.1	76.7	73.5	80.3
λ = 0.001	82.4	82.2	77.0	73.9	71.1	77.3
λ = 0.01	85.7	84.3	77.7	75.7	70.5	78.8
λ = 0.1	85.7	80.1	76.3	73.1	68.6	76.8

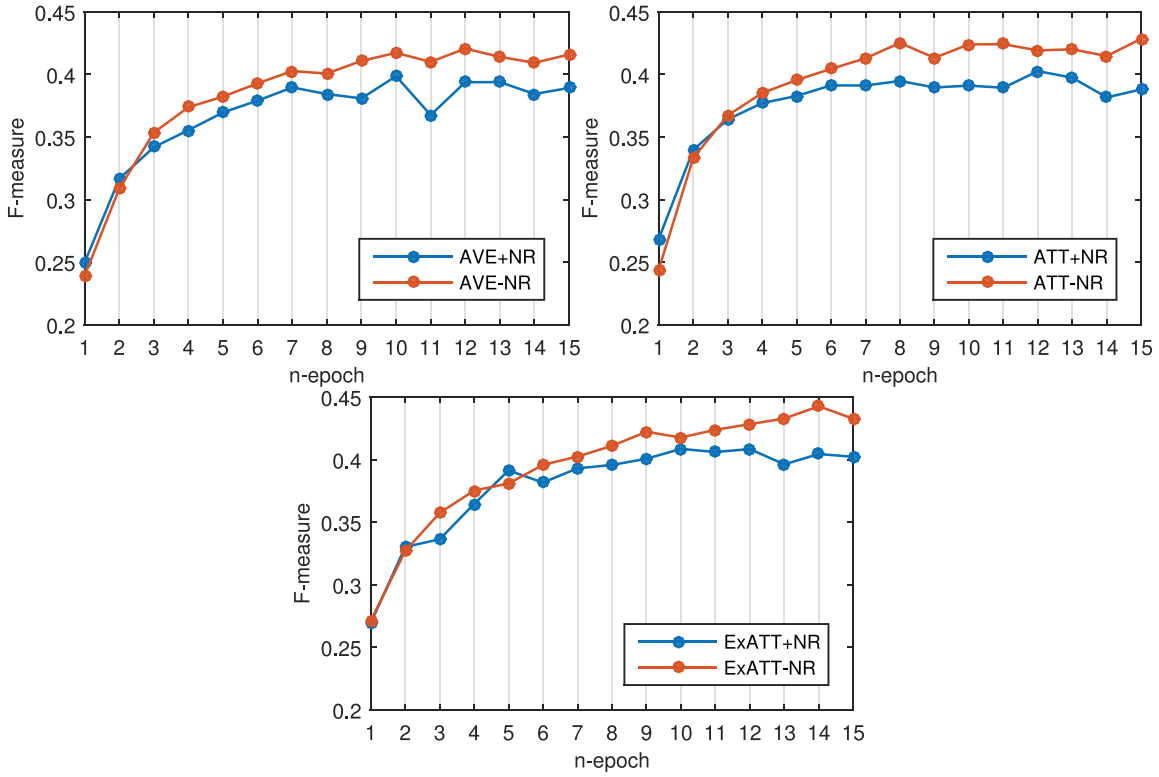

Fig. 8. Impact of NR for model convergence. "+NR" means not relieving NR impact with λ of 1; "−NR" is opposite with λ of 0. ExATT is based on the loss function of Variant-3.

5. Conclusion and future works

In this work, we propose a ranking based cost-sensitive multi-label learning for distant relation extraction aiming to leverage class ties to enhance relation extraction and relieving class imbalance problem. To exploit class ties between relations to improve relation extraction, we propose a general ranking based multi-label learning framework combined with convolutional neural networks, in which ranking based loss functions with regularization technique are introduced to learn the latent connections between relations. Furthermore, to deal with the problem of *class imbalance* in distant supervision relation extraction, we further adopt cost-sensitive learning to rescale the costs from the positive and negative labels. In the experimental study, we further do experiments to analyze the effectiveness of our novel cost-sensitive ranking loss functions. The evaluation experiments on the effectiveness of regularization have further be conducted.

In the future, we will focus on the following aspects: (1) Our method in this paper considers pairwise intersections between labels, so to better exploit class ties, we will extend our method to exploit all other labels' influences on each relation for relation extraction, transferring *second-order* to *high-order* (Zhang & Zhou, 2014); (2) We will regard the task of distant supervision relation extraction as a multi-instance based learning-to-rank problem, and will take the view from learning-to-rank to design the algorithms and combine other advanced tricks from information retrieval field; (3) What effects will entity pairs take to the relation extraction performance? Can we use a general entity pair replacement (e_1, e_2) to represent all entity pairs? Answering the two problems may help the transfer learning of RE systems.

Acknowledgments

This work was supported by the National High-tech Research and Development Program (863 Program) (No. 2014AA015105) and National Natural Science Foundation of China (No. 61602490).

Supplementary material

Supplementary material associated with this article can be found, in the online version, at 10.1016/j.ipm.2019.102096.

References

Bahdanau, D., Cho, K., & Bengio, Y. (2015). *Neural machine translation by jointly learning to align and translate. Proceedings of ICLR* http://arxiv.org/abs/1409.0473.
Bollacker, K. D., Evans, C., Paritosh, P., Sturge, T., & Taylor, J. (2008). *Freebase: A collaboratively created graph database for structuring human knowledge. Proceedings of KDD*1247–1250.
Chen, L., Feng, Y., Huang, S., Luo, B., & Zhao, D. (2018). Encoding implicit relation requirements for relation extraction: A joint inference approach. *Artificial Intelligence, 265*, 45–66.
Evgeniou, T., Micchelli, C. A., & Pontil, M. (2005). Learning multiple tasks with kernel methods. *Journal of Machine Learning Research, 6*(Apr), 615–637.
Feng, J., Huang, M., Zhao, L., Yang, Y., & Zhu, X. (2018). *Reinforcement learning for relation classification from noisy data. Proceedings of the thirty-second AAAI conference on artificial intelligence, (AAAI-18), the 30th innovative applications of artificial intelligence (IAAI-18), and the 8th AAAI symposium on educational advances in artificial intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2–7, 2018*.
Fürnkranz, J., Hüllermeier, E., Mencía, E. L., & Brinker, K. (2008). Multilabel classification via calibrated label ranking. *Machine Learning, 73*(2), 133–153.
Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... Bengio, Y. (2014). *Generative adversarial nets. Advances in neural information processing systems*.
Han, X., Liu, Z., & Sun, M. (2018). Denoising distant supervision for relation extraction via instance-level adversarial training. *CoRR* arXiv:1805.10959.
Han, X., & Sun, L. (2016). *Global distant supervision for relation extraction. Proceedings of AAAI*.
He, H., & Garcia, E. A. (2009). Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering, 21*(9), 1263–1284.
Hoffmann, R., Zhang, C., Ling, X., Zettlemoyer, L., & Weld, D. S. (2011). *Knowledge-based weak supervision for information extraction of overlapping relations. Proceedings of ACL-HLT*.
Huang, C., Li, Y., Change Loy, C., & Tang, X. (2016). *Learning deep representation for imbalanced classification. Proceedings of the IEEE conference on computer vision and pattern recognition*.
Japkowicz, N., & Stephen, S. (2002). The class imbalance problem: A systematic study. *Intelligent Data Analysis, 6*(5), 429–449.
Ji, G., Liu, K., He, S., & Zhao, J. (2017). *Distant supervision for relation extraction with sentence-level attention and entity descriptions. AAAI*3060–3066.
Jiang, X., Ye, H., Luo, Z., Chao, W., & Ma, W. (2018). *Interpretable rationale augmented charge prediction system. The 27th international conference on computational linguistics: System demonstrations*.
Khan, S. H., Bennamoun, M., Sohel, F., & Togneri, R. (2015). Cost sensitive learning of deep feature representations from imbalanced data. *CoRR* arXiv:1508.03422.
LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature, 521*(7553), 436–444.
Lin, Y., Liu, Z., & Sun, M. (2017). *Neural relation extraction with multi-lingual attention. Proceedings of association for computational linguistics*.
Lin, Y., Shen, S., Liu, Z., Luan, H., & Sun, M. (2016). *Neural relation extraction with selective attention over instances. Proceedings of ACL*.
Liu, T., Wang, K., Chang, B., & Sui, Z. (2017). *A soft-label method for noise-tolerant distantly supervised relation extraction. Proceedings of empirical methods in natural language processing*.
Liu, T., Zhang, X., Zhou, W., & Jia, W. (2018). *Neural relation extraction via inner-sentence noise reduction and transfer learning. Proceedings of empirical methods in natural language processing*.
Liu, T.-Y. (2009). Learning to rank for information retrieval. *Foundations and Trends in Information Retrieval, 3*(3), 225–331.
Luo, B., Feng, Y., Wang, Z., Zhu, Z., Huang, S., Yan, R., & Zhao, D. (2017). *Learning with noise: Enhance distantly supervised relation extraction with dynamic transition matrix. Proceedings of association for computational linguistics*.
Luong, T., Pham, H., & Manning, C. D. (2015). *Effective approaches to attention-based neural machine translation. Proceedings of EMNLP*.
Mintz, M., Bills, S., Snow, R., & Jurafsky, D. (2009). *Distant supervision for relation extraction without labeled data. Proceedings of ACL-IJCNLP*.
Qin, P., Xu, W., & Wang, W. Y. (2018). Dsgan: Generative adversarial training for distant supervision relation extraction. *CoRR* arXiv:1805.09929.
Qin, P., Xu, W., & Wang, W. Y. (Xu, Wang, 2018b). *Robust distant supervision relation extraction via deep reinforcement learning. Proceedings of the 56th annual meeting of the association for computational linguistics, ACL 2018, Melbourne, Australia, July 15–20, 2018, volume 1: Long papers*.
Riedel, S., Yao, L., & McCallum, A. (2010). *Modeling relations and their mentions without labeled text. Proceedings of ECML-PKDD*. Springer148–163.
Santos, C. N.d., Xiang, B., & Zhou, B. (2015). *Classifying relations by ranking with convolutional neural networks. Proceeding of ACL*.
Severyn, A., & Moschitti, A. (2015). *Learning to rank short text pairs with convolutional deep neural networks. Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval*. ACM373–382.
Shen, W., Wang, X., Wang, Y., Bai, X., & Zhang, Z. (2015). *Deepcontour: A deep convolutional feature learned by positive-sharing loss for contour detection. Proceedings of the IEEE conference on computer vision and pattern recognition*.
Surdeanu, M., Tibshirani, J., Nallapati, R., & Manning, C. D. (2012). *Multi-instance multi-label learning for relation extraction. Proceedings of EMNLP*.
Wang, L., Cao, Z., de Melo, G., & Liu, Z. (2016). *Relation classification via multi-level attention CNNs. Proceedings of ACL, volume 1: Long papers*.
Weston, J., Bengio, S., & Usunier, N. (2011). *WSABIE: Scaling up to large vocabulary image annotation. Proceedings of IJCAI*.
Ye, H., Chao, W., Luo, Z., & Li, Z. (2017). *Jointly extracting relations with class ties via effective deep ranking. Proceedings of association for computational linguistics*.
Ye, H., Jiang, X., Luo, Z., & Chao, W. (2018). Interpretable charge predictions for criminal cases: Learning to generate court views from fact descriptions. *CoRR*. arXiv:1802.08504.
Ye, H., Li, W., & Wang, L. (2019). Jointly learning semantic parser and natural language generator via dual information maximization. *CoRR*. arXiv:1906.00575.
Ye, H., & Wang, L. (2018). *Semi-supervised learning for neural keyphrase generation. Proceedings of empirical methods in natural language processing*.
Ye, H., Yan, Z., Luo, Z., & Chao, W. (2017). *Dependency-tree based convolutional neural networks for aspect term extraction. Advances in knowledge discovery and data mining - 21st Pacific-Asia conference, PAKDD 2017, Jeju, South Korea, May 23–26, 2017, proceedings, part II*.
Yu Mo, M. G., & Dredze, M. (2014). *Factor-based compositional embedding models. NIPS workshop on learning semantics*.
Zeng, D., Liu, K., Chen, Y., & Zhao, J. (2015). *Distant supervision for relation extraction via piecewise convolutional neural networks. Proceedings of EMNLP*.
Zeng, D., Liu, K., Lai, S., Zhou, G., Zhao, J., et al. (2014). *Relation classification via convolutional deep neural network. Proceeding of COLING*.
Zeng, W., Lin, Y., Liu, Z., & Sun, M. (2016). Incorporating relation paths in neural relation extraction. *CoRR* arXiv:1609.07479.
Zhang, M.-L., & Zhou, Z.-H. (2006). Multilabel neural networks with applications to functional genomics and text categorization. *IEEE Transactions on Knowledge and Data Engineering, 18*(10), 1338–1351.
Zhang, M.-L., & Zhou, Z.-H. (2014). A review on multi-label learning algorithms. *IEEE Transactions on Knowledge and Data Engineering, 26*(8), 1819–1837.
Zhao, F., Huang, Y., Wang, L., & Tan, T. (2015). *Deep semantic ranking based hashing for multi-label image retrieval. Proceedings of CVPR*.
Zheng, H., Li, Z., Wang, S., Yan, Z., & Zhou, J. (2016). *Aggregating inter-sentence information to enhance relation extraction. Thirtieth AAAI conference on artificial intelligence*.
Zhou, Z.-H., Zhang, M.-L., Huang, S.-J., & Li, Y.-F. (2012). Multi-instance multi-label learning. *Artificial Intelligence, 176*(1), 2291–2320.