Revisiting the Negative Data of Distantly Supervised Relation Extraction

Chenhao Xie[1,2], **Jiaqing Liang**[1,2], **Jingping Liu**[1], **Chengsong Huang**[1], **Wenhao Huang**[1], **Yanghua Xiao**[1,3*]
[1]Shanghai Key Laboratory of Data Science, School of Computer Science, Fudan University
[2]Shuyan Technology Inc., Shanghai, China
[3]Fudan-Aishu Cognitive Intelligence Joint Research Center, Shanghai, China
{redreamality, l.j.q.light}@gmail.com
{jpliu17, huangcs19, whhuang17, shawyh}@fudan.edu.cn

Abstract

Distantly supervision automatically generates plenty of training samples for relation extraction. However, it also incurs two major problems: noisy labels and imbalanced training data. Previous works focus more on reducing wrongly labeled relations (false positives) while few explore the missing relations that are caused by incompleteness of knowledge base (false negatives). Furthermore, the quantity of negative labels overwhelmingly surpasses the positive ones in previous problem formulations. In this paper, we first provide a thorough analysis of the above challenges caused by negative data. Next, we formulate the problem of relation extraction into as a positive unlabeled learning task to alleviate false negative problem. Thirdly, we propose a pipeline approach, dubbed RERE, that first performs sentence classification with relational labels and then extracts the subjects/objects. Experimental results show that the proposed method consistently outperforms existing approaches and remains excellent performance even learned with a large quantity of false positive samples. Source code is available online[1].

1 Introduction

Relational extraction is a crucial step towards knowledge graph construction. It aims at identifying relational triples from a given sentence in the form of ⟨subject, relation, object⟩, in short, $\langle s, r, o \rangle$. For example, given S1 in Figure 1, we hope to extract ⟨WILLIAM SHAKESPEARE, BIRTHPLACE, STRATFORD-UPON-AVON⟩.

This task is usually modeled as a supervised learning problem and *distant supervision* (Mintz et al., 2009) is utilized to acquire large-scale training data. The core idea is to obtain training data

Knowledge base

Subject	Relation	Object
William Shakespeare	Birthplace	Stratford-upon-Avon
Thomas Cruise	Birthplace	New York
Joe Biden	Birthplace	Scranton

align and label

Corpus

S1	**William Shakespeare** was born and raised in **Stratford-upon-Avon**.	Birthplace (TP)
S2	**Shakespeare** spent the last few years of his life in **Stratford-upon-Avon**.	Birthplace (FP)
S3	**Buffett** was born in 1930 in **Omaha, Nebraska**.	NA (R-FN)
S4	**Thomas Cruise Mapother IV** was born in **Syracuse, New York**.	Birthplace (E-PL)
S5	**William Swartz** was born in **Scranton**, which is also the birth place of US President **Joe Biden**.	Birthplace (E-FN)

Figure 1: Illustration of distant supervision process. S2-S5 are examples for four kinds of label noise. TP, FP, FN and PL mean *true positive, false positive, false negative* and *partially labeled*, respectively. "R-" or "E-" indicates whether the error occurs at *relation-level* or *entity-level*. **Bold tokens** are ground-truth subjects/objects. Underlined tokens together with the relation in the third column are labeled by distant supervision. "NA" means no relation.

is through automatically labeling a sentence with existing relational triples from a knowledge base (KB). For example, given a triple $\langle s, r, o \rangle$ and a sentence, if the sentence contains both s and o, distant supervision methods regard $\langle s, r, o \rangle$ as a valid sample for the sentence. If no relational triples are applicable, the sentence is labeled as "NA".

Despite the abundant training data obtained with distant supervision, nonnegligible errors also occur in the labels. There are two types of errors. In the first type, the labeled relation does not conform with the original meaning of sentence, and this type of error is referred to as *false positive* (FP). For example, in *S2*, the sentence "*Shakespeare spent the last few years of his life in Stratford-upon-Avon.*" does not express the relation BIRTHPLACE, thus being a FP. In the second type, large amounts of

*Corresponding author
[1]https://github.com/redreamality/RERE-relation-extraction

Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, pages 3572–3581
August 1–6, 2021.

relations in sentences are missing due to the incompleteness of KB, which is referred to as *false negative* (FN). For instance, in *S3*, "*Buffett was born in 1930 in Omaha, Nebraska.*" is wrongly labeled as NA since there is no relation (e.g., BIRTHPLACE) between BUFFETT and OMAHA, NEBRASKA in the KB. Many efforts have been devoted to solving the FP problem, including pattern-based methods (Jia et al., 2019), multi-instance learning methods (Lin et al., 2016; Zeng et al., 2018a) and reinforcement learning methods (Feng et al., 2018). Significant improvements have been made.

However, FN problem receives much less attention (Min et al., 2013; Xu et al., 2013; Roller et al., 2015). To the best of our knowledge, none existing work with deep neural networks to solve this problem. We argue that this problem is fatal in practice since there are massive FN cases in datasets. For example, there exist at least 33% and 35% FNs in NYT and SKE datasets, respectively. We will deeply analyze the problem in Section 2.1

Another huge problem in relation extraction is the *overwhelming negative labels*. As is widely acknowledged, information extraction tasks are highly imbalanced in class labels (Chowdhury and Lavelli, 2012; Lin et al., 2018; Li et al., 2020). In particular, the negative labels account for most of the labels in relation extraction under almost any problem formulation, which makes relation extraction a hard machine learning problem. We systematically analyze this in Section 2.2.

In this paper, we address these challenges caused by negative data. Our main contribution can be summarized as follows.

- We systematically compare the class distributions of different problem modeling and explain why first extract relation then entities, i.e., the third paradigm (P3) in Section 2.2, is superior to the others.
- Based on the first point, we adopt P3 and propose a novel two-staged pipeline model dubbed RERE. It first detects relation at sentence level and then extracts entities for a specific relation. We model the false negatives in relation extraction as "unlabeled positives" and propose a multi-label collective loss function.
- Our empirical evaluations show that the proposed method consistently outperforms existing approaches, and achieves excellent performance even learned with a large quantity of false positive samples. We also provide two carefully annotated test sets aiming at reducing the false negatives of previous annotation, namely, NYT21 and SKE21, with 370 and 1150 samples, respectively.

2 Problem Analysis and Pilot Experiments

We use $(\mathbf{c}_i, T_i)$ to denote a training instance, where $\mathbf{c}_i$ is a sentence consisting of N tokens $\mathbf{c}_i = [c_{i1}, ..., c_{iN}]$ labeled by a set of triples $T_i = \{\langle s, r, o\rangle\}$ from the training set $\mathcal{D}$. For rigorous definition, $[c_{i1}, ..., c_{iN}]$ can be viewed as an ordered set $\{(c_{i1}, 1), ..., (c_{iN}, N)\}$ so that set operations can be applied. We assume $r \in \mathcal{R}$, where $\mathcal{R}$ is a finite set of all relations in $\mathcal{D}$. Other model/task-specific notations are defined after each problem formulation.

We now clarify some terms used in the introduction and title without formal definition. A **negative sample** refers to a triple $t \notin T_i$. **Negative label** refers to the negative class label (e.g., usually "0" for binary classification), used for supervision with respect to task-specific models. Under different task formulation, the negative labels can be different. **Negative data** is a general term that includes both negative labels and negative samples. There are two kinds of *false negatives*. **Relation-level false negative** (S3 in Figure 1) refers to the situation where there exists $t' = \langle s', r', o'\rangle \notin T_i$, but r' is actually expressed by $\mathbf{c}_i$, and does not appear in other $t \in T_i$. Similarly, **Entity-level false negative** (S4 and S5 in Figure 1) means r' appears in other $t \in T_i$. **Imbalanced class distribution** means that the quantity of negative labels is much larger than that of positive ones.

2.1 Addressing the False Negatives

As shown in Table 1, the triples in NYT (SKE) datasets[2] labeled by Freebase[3] (BaiduBaike[4]) is 88,253 (409,767), while the ones labeled by Wikidata[5] (CN-DBPedia[6]) are 58,135 (342,931). In other words, there exists massive FN matches if only labeled by one KB due to the incompleteness of KBs. Notably, we find that the FN rate is underestimated by previous researches (Min

[2]Detailed description of datasets is in Sec. 5.1
[3](Bollacker et al., 2008)
[4]https://baike.baidu.com/
[5](Vrandecic and Krötzsch, 2014)
[6] (Xu et al., 2017)

et al., 2013; Xu et al., 2013), based on the manual evaluation of which there are 15%-35% FN matches. This discrepancy may be caused by human error. In specific, a volunteer may accidentally miss some triples. For example, as pointed out by Wei et al. (2020, in Appendix C), the test set of NYT11 (Hoffmann et al., 2011) missed lots of triples, especially when multiple relations occur in a same sentence, though labeled by human. That also provides an evidence that FN's are harder to discover than FP's.

	NYT (English)		SKE (Chinese)	
# Sentence	56,196		194,747	
	# Triples	# Rels	# Triples	# Rels
Original	88,253	23	409,767	49
Re-labeled	58,135	57	342,931	378
Intersection	13,848	18	121,326	46
Union	132,540	62	631,372	381
Original FNR	≥ 0.33		≥ 0.35	
Relabel FNR	≥ 0.56		≥ 0.46	

Table 1: Statistics of the quantity of distantly labeled relational triples by using different KB's. The "original" refers to freebase for NYT and BaiduBaike for SKE. The "relabeled" means aligning using Wikidata and CN-DBpedia to re-label NYT and SKE datasets. In specific, we consider triples with the same subject and object to be candidate triples and use a relation mapping table to determine whether the triples match. The intersection of SKE dataset has two values because the original relation has a one-to-many mapping with relations in CN-DBpedia. FNR stands for false negative rates, calculated by using the # Triples in Original (Re-labeled) divided by the union.

2.2 Addressing the Overwhelming Negative Labels

We point out that some of the previous paradigms designed for relation extraction aggravate the imbalance and lead to inefficient supervision. The mainstream approaches for relation extraction mainly fall into three paradigms depending on what to extract first.

P1 The first paradigm is a pipeline that begins with named entity recognition (NER) and then classifies each entity pair into different relations, i.e., [s, o then r]. It is adopted by many traditional approaches (Mintz et al., 2009; Chan and Roth, 2011; Zeng et al., 2014, 2015; Gormley et al., 2015; dos Santos et al., 2015; Lin et al., 2016).

P2 The second paradigm first detects all possible subjects in a sentence then identifies objects with respect to each relation, i.e., [s then r, o]. Specific implementation includes modeling relation extraction as multi-turn question answering (Li et al., 2019), span tagging (Yu et al., 2020) and cascaded binary tagging (Wei et al., 2020).

P3 The third paradigm first perform sentence-level relation detection (cf. P1, which is at entity pair level.) then extract subjects and entities, i.e., [r then s, o]. This paradigm is largely unexplored. HRL (Takanobu et al., 2019) is hitherto the only work to apply this paradigm based on our literature review.

We provide theoretical analysis of the output space and class prior with statistical support from three datasets (see Section 5.1 for description) of the three paradigms in Table 2. The second step of P1 can be compared with the first step of P3. Both of them find relation from a sentence (P1 with target entity pair given). Suppose a sentence contains m entities[7], the classifier has to decide relation from $\mathcal{O}(m^2)$ entity pairs, while in reality, relations are often sparse, i.e., $\mathcal{O}(m)$. In other words, most entity pairs in P1 do not form valid relation, thus resulting in a low class prior. The situation is even worse when the sentence contains more entities, such as in NYT11-HRL. For P2, we demonstrate with the problem formulation of CASREL (Wei et al., 2020). The difference of the first-step class prior between P2 and P3 depends on the result of comparison between # relations and average sentence length (i.e., $|\mathcal{R}|$ and $\bar{N}$), which varies in different scenarios/domains. However, π_2 of P2 is extremely low, where a classifier has to decide from a space of $|\mathcal{R}| * \bar{N}$. In contrast, P3 only need to decide from $4 * \bar{N}$ based on our task formulation (Section 3.1)

Other task formulations include jointly extracting the relation and entities (Yu and Lam, 2010; Li and Ji, 2014; Miwa and Sasaki, 2014; Gupta et al., 2016; Katiyar and Cardie, 2017; Ren et al., 2017) and recently in the manner of sequence tagging (Zheng et al., 2017), sequence-to-sequence learning (Zeng et al., 2018b). In contrast to the aforementioned three paradigms, most of these methods actually provide an *incomplete decision space* that cannot handle all the situation of relation extrac-

[7] Below the same.

Paradigm	Theoretical		NYT10-HRL		NYT11-HRL		SKE	
			$\lvert\mathcal{R}\rvert$=31,$\bar{N}$= 39.08		$\lvert\mathcal{R}\rvert$=11, $\bar{N}$=39.46		$\lvert\mathcal{R}\rvert$=51,$\bar{N}$= 54.67	
	π_1	π_2	π_1	π_2	π_1	π_2	π_1	π_2
s, o then r	–	$\mathbb{E}[\frac{\sum y}{\lvert\mathcal{R}\rvert}]$	–	0.01421	–	0.00280	–	0.00494
s then r, o	$\mathbb{E}[\frac{\sum y}{N}]$	$\mathbb{E}[\frac{\sum y}{N*\lvert\mathcal{R}\rvert}]$	0.0585	0.00093	0.0574	0.00257	0.0405	0.00067
r then s, o	$\mathbb{E}[\frac{\sum y}{\lvert\mathcal{R}\rvert}]$	$\mathbb{E}[\frac{\sum y}{4*N}]$	0.0390	0.00842	0.0826	0.00835	0.0344	0.00927

Table 2: Comparison of class prior under different relation extraction paradigms. $|\mathcal{R}|$ means the total number of relations and $\bar{N}$ is the average sentence length. π_1 (π_2) refers to the class prior for the first (second) task in the pipeline. π_1 for the first paradigm is omitted because it is often considered a preceding step. $\sum y$ is the summation of 1's in labels, of using which our intention is to represent the information a positive sample conveys.

tion, for example, the overlapping one (Wei et al., 2020).

3 Solution Framework

3.1 Framework of RERE

Given an instance (c_i, T_i) from $\mathcal{D}$, the goal of training is to maximize the likelihood defined in Eq. (1). It is decomposed into two components by applying the definition of conditional probability, formulated in Eq. (2).

$$\prod_{i=1}^{|\mathcal{D}|} \Pr(T_i|\mathbf{c}_i;\theta) \tag{1}$$

$$= \prod_{i=1}^{|\mathcal{D}|} \prod_{r\in T_i} \Pr(r|\mathbf{c}_i;\theta) \prod_{\langle s,o\rangle \in T_i|r} \Pr(s,o|r,\mathbf{c}_i;\theta), \tag{2}$$

where we use $r \in T_i$ as a shorthand for $r \in \{r \mid \langle s, r, o\rangle \in T_i\}$, which means that r occurs in the triple set w.r.t. $\mathbf{c}_i$; Similarly, $s \in T_i$, $\langle s, o\rangle \in T_i|r$ stands for $s \in \{s \mid \langle s, r, o\rangle \in T_i|r\}$ and $\langle s, o\rangle \in \{\langle s, o\rangle \mid \langle s, r, o\rangle \in T_i|r\}$, respectively. $T_i|r$ represents a subset of T_i with a common relation r. $\mathbb{1}[\cdot]$ is an indicator function; $\mathbb{1}[\text{condition}] = 1$ when the condition happens. We denote by θ the model parameters. Under this decomposition, relational triple extraction task is formulated into two subtasks: *relation classification* and *entity extraction*.

Relation Classification. As is discussed, building relation classifier at entity-pair level will introduce excessive negative samples and form a hard learning problem. Therefore, we alternatively model the relation classification at sentence level. Intuitively speaking, we hope that the model could capture *what relation a sentence is expressing*. We formalize it as a multi-label classification task.

$$\Pr(r|\mathbf{c}_i;\theta) = \prod_{j=1}^{|\mathcal{R}|} (\hat{y}^j_{rc})^{\mathbb{1}[y^j_{rc}=1]} (1-\hat{y}^j_{rc})^{\mathbb{1}[y^j_{rc}=0]}, \tag{3}$$

where $\hat{y}^j_{rc}$ is the probability that $\mathbf{c}$ is expressing r_j, the j-th relation[8]. y^j_{rc} is the ground truth from the labeled data; $y^j_{rc} = 1$ is equivalent to $r_j \in T_i$ while $y^j_{rc} = 0$ means the opposite.

Entity Extraction. We then model entity extraction task. We observe that given the relation r and context $\mathbf{c}_i$, it naturally forms a machine reading comprehension (MRC) task (Chen, 2018), where $(r, \mathbf{c}_i, s/o)$ naturally fits into the paradigm of (QUERY, CONTEXT, ANSWER). Particularly, the subjects and objects are continuous spans from $\mathbf{c}_i$, which falls into the category of *span extraction*. We adopt the boundary detection model with answer pointer (Wang and Jiang, 2017) as the output layer, which is widely used in MRC tasks. Formally, for a sentence of N tokens,

$$\Pr(s,o|r,\mathbf{c}_i;\theta) = \prod_{k\in\mathcal{K}} \prod_{n=1}^{N} (\hat{y}^{n,k}_{ee})^{\mathbb{1}[y^{n,k}_{ee}=1]} (1-\hat{y}^{n,k}_{ee})^{\mathbb{1}[y^{n,k}_{ee}=0]}, \tag{4}$$

where $\mathcal{K} = \{s_{start}, s_{end}, o_{start}, o_{end}\}$ represents the identifier of each pointer; $\hat{y}^{n,k}_{ee}$ refers to the probability of n-th token being the start/end of the subject/object. $y^{n,k}_{ee}$ is the ground truth from the training data; if $\exists s \in T_i|r$ occurs in $\mathbf{c}_i$ at position from n to $n+l$, then $y^{n,s_{start}}_{ee} = 1$ and $y^{n+l,s_{end}}_{ee} = 1$, otherwise 0; the same applies for the *objects*.

[8] $\hat{y}^j_{rc}$ is parameterized by θ, omitted in the equation for clarity, below the same.

3.2 Advantages

Our task formulation shows several advantages. By adopting P3 as paradigm, the first and foremost advantage of our solution is that it suffers less from the imbalanced classes (Section 2.2). Secondly, relation-level false negative is easy to recover. When modeled as a standard classification problem, many off-the-shelf methods on positive unlabeled learning can be leveraged. Thirdly, entity-level false negatives do not affect relation classification. Taking S5 in Figure 1 as an example, even though the BIRTHPLACE relation between WILLIAM SWARTZ and SCRANTON is missing, the relation classifier can still capture the signal from the other sample with a same relation, i.e., ⟨ JOE BIDEN, BIRTHPLACE, SCRANTON ⟩. Fourthly, this kind of modeling is easy to update with new relations without the need of retraining a model from bottom up. Only relation classifier needs to be redesigned, while entity extractor can be updated in an online manner without modifying the model structure. Last but not the least, relation classifier can be regarded as a pruning step when applied to practical tasks. Many existing methods treat relation extraction as question answering (Li et al., 2019; Zhao et al., 2020). However, without first identifying the relation, they all need to iterate over all the possible relations and ask diverse questions. This results in extremely low efficiency where time consumed for predicting one sample may take up to $|\mathcal{R}|$ times larger than our method.

4 Our Model

The relational triple extraction task decomposed in Eq. (2) inspires us to design a two-staged pipeline, in which we first detect relation at sentence level and then extract subjects/objects for each relation. The overall architecture of RERE is shown in Figure 2.

4.1 Sentence Classifier with Relational Label

We first detect relation at sentence level. The input is a sequence of tokens $\mathbf{c}$ and we denote by $\hat{\mathbf{y}}_{rc} = [\hat{y}^1_{rc}, \hat{y}^2_{rc}, ..., \hat{y}^{|\mathcal{R}|}_{rc}]$ the output vector of the model, which aims to estimate $\hat{y}^i_{rc}$ in Eq. (3). We use BERT (Devlin et al., 2019) for English and RoBERTa (Liu et al., 2019) for Chinese, pre-trained language models with multi-layer bidirectional Transformer structure (Vaswani et al., 2017), to encode the inputs[9]. Specifically, the input sequence $\mathbf{x}_{rc} = [\texttt{[CLS]}, \mathbf{c}_i, \texttt{[SEP]}]$, which is fed into BERT for generating a token representation matrix $\mathbf{H}_{rc} \in \mathbb{R}^{N \times d}$, where d is the hidden dimension defined by pre-trained Transformers. We take $\mathbf{h}^0_{rc}$, which is the encoded vector of the first token `[CLS]`, as the representation of the sentence. The final output of relation classification module $\hat{\mathbf{y}}_{rc}$ is defined in Eq. (5).

$$\hat{\mathbf{y}}_{rc} = \sigma(\mathbf{W}_{rc}\mathbf{h}^0_{rc} + \mathbf{b}_{rc}), \tag{5}$$

where $\mathbf{W}_{rc}$ and $\mathbf{b}_{rc}$ are trainable model parameters, representing weights and bias, respectively; σ denotes the sigmoid activation function.

4.2 Relation-specific Entity Extractor

After the relation detected at sentence-level, we extract subjects and objects for each candidate relation. We aim to estimate $\hat{\mathbf{y}}_{ee} = [0, 1]^{N \times 4}$, of which each element corresponds to $\hat{y}^{n,k}_{ee}$ in Eq. (4), using a deep neural model. We take $\hat{\mathbf{y}}_{rc}$, the one-hot output vector of relation classifier, and generate query tokens $\mathbf{q}$ using each of the detected relations (i.e., the "1"s in $\hat{\mathbf{y}}_{rc}$). We are aware that many recent works (Li et al., 2019; Zhao et al., 2020) have studied how to generate diverse queries for the given relation, which have the potential of achieving better performance. Nevertheless, that is beyond the scope of this paper. To keep things simple, we use the *surface text* of a relation as the query.

Next, the input sequence is constructed as $\mathbf{x}_{ee} = [\texttt{[CLS]}, \mathbf{q}_i, \texttt{[SEP]}, \mathbf{c}_i, \texttt{[SEP]}]$. Like Section 4.1, we get the token representation matrix $\mathbf{H}_{ee} \in \mathbb{R}^{N \times d}$ from BERT. The k-th output pointer of entity extractor is defined by

$$\hat{\mathbf{y}}^k_{ee} = \sigma(\mathbf{W}^k_{ee}\mathbf{H}_{ee} + \mathbf{b}^k_{ee}), \tag{6}$$

where $k \in \{s_{start}, s_{end}, o_{start}, o_{end}\}$ is in accordance to Eq. (4); $\mathbf{W}^k_{ee}$ and $\mathbf{b}^k_{ee}$ are the corresponding parameters.

The final subject/object spans are generated by pairing the nearest s_{start}/o_{start} with s_{end}/o_{end}. Next, all subjects are paired to the nearest object. If multiple objects occur before the next subject appears, all subsequent objects will be paired with it until next subject occurs.

[9]For convenience, we refer to the pre-trained Transformer as BERT hereinafter.

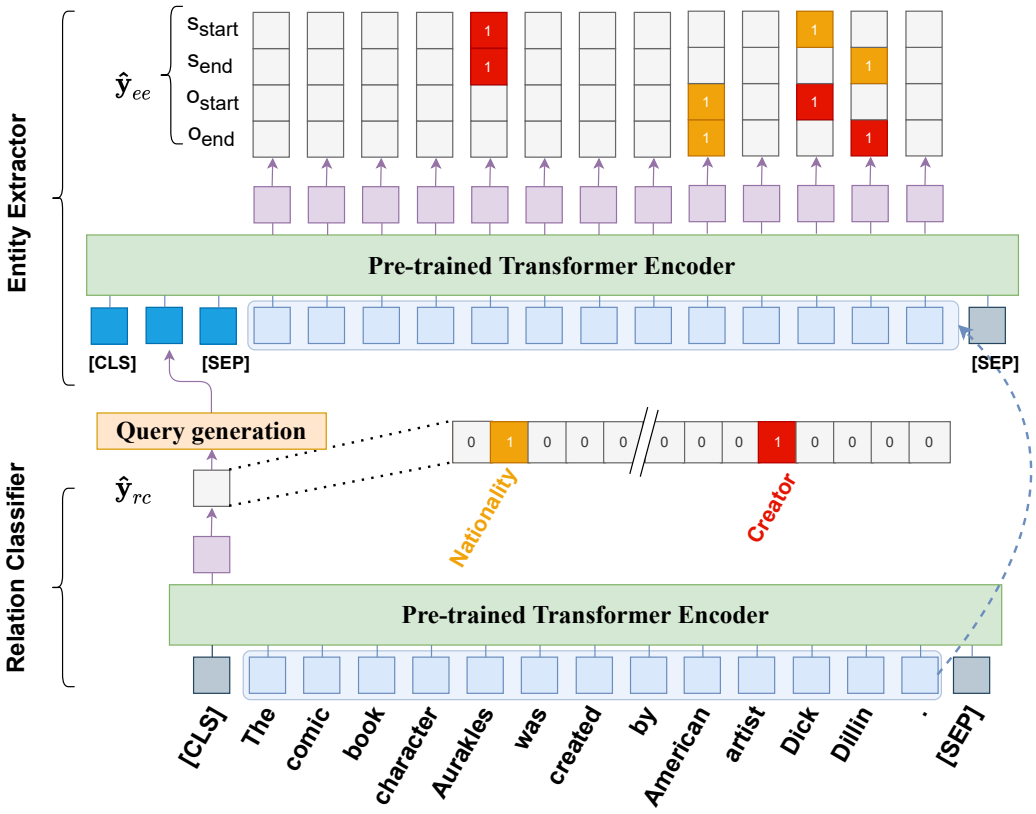

Figure 2: The overall architecture of ReRe. In this example, there are two relations, NATIONALITYand CREATOR, can be found in the Relation Classifier, which will be sent to the Entity Extractor one by one along with the sentence. When The relation NATIONALITY is extracted, the Entity Extractor will find the position of the subject and object of Nationality. The word AMERICAN and DICK DILLIN will be found. The relation CREATOR will then be handled similarly. The values of grey blocks in $\hat{\mathbf{y}}_{ee}$ are zero.

4.3 Multi-label Collective Loss function

In normal cases, the log-likelihood is taken as the learning objective. However, as is emphasized, there exist many false negative samples in the training data. Intuitively speaking, the negative labels cannot be simply considered as negative. Instead, a small portion of the negative labels should be considered as unlabeled positives and their influence towards the penalty should be eradicated. Therefore, we adopt cPU (Xie et al., 2020), a collective loss function that is designed for *positive unlabeled learning* (PU learning). To briefly review, cPU considers the learning objective to be the *correctness* under a surrogate function,

$$\ell(\hat{y}, y) = \ln(c(\hat{y}, y)), \tag{7}$$

where they redefine the *correctness* function for PU learning as

$$c(\hat{y}, y) = \begin{cases} \mathbb{E}[\hat{y}] & \text{if } y = 1, \\ 1 - |\mathbb{E}[\hat{y}] - \mu| & \text{otherwise,} \end{cases} \tag{8}$$

where μ is the ratio of false negative data (i.e., the unlabeled positive in the original paper).

We extend it to multi-label situation by embodying the original expectation at sample level. Due to the fact that class labels are highly imbalanced for our tasks, we introduce a class weight $\gamma \in (0, 1)$ to downweight the positive penalty. For relation classifier,

$$\ell_{rc}(\hat{\mathbf{y}}, \mathbf{y}) = \begin{cases} -\gamma_{rc} \ln(\frac{1}{|\mathcal{R}|} \sum_{i=1}^{|\mathcal{R}|} \hat{y}^i_{rc}]) & \text{if } y^i_{rc} = 1 \\ -\ln(1 - |\frac{1}{|\mathcal{R}|} \sum_{i=1}^{|\mathcal{R}|} \hat{y}^i_{rc} - \mu_{rc}|) & \text{otherwise.} \end{cases} \tag{9}$$

For entity extractor,

$$\ell_{ee}(\hat{\mathbf{y}}^k, \mathbf{y}^k) = \begin{cases} -\gamma_{ee} \ln(\sum_{n=1}^{N} \hat{y}^{n,k}_{ee}]) & \text{if } y^{n,k}_{ee} = 1 \\ -\ln(1 - |\sum_{n=1}^{N} \hat{y}^{n,k}_{ee} - \mu_{ee}|) & \text{otherwise.} \end{cases} \tag{10}$$

In practice, we set $\mu = \pi(\tau + 1)$, where $\tau \approx 1 - \frac{\text{\# labeled positive}}{\text{\# all positive}}$ is the ratio of false negative and π is the class prior. Note that μ is not difficult to estimate for both relation classification and entity extraction task in practice. Besides various

of methods in the PU learning (du Plessis et al., 2015; Bekker and Davis, 2018) for estimating it, an easy approximation is $\mu \approx \pi$ when $\pi \ll \tau$, which happens to be the case for our tasks.

5 Experiments

5.1 Datasets

Our experiments are conducted on these four datasets[10]. Some statistics of the datasets are provided in Table 1 and Table 2. In relation extraction, some datasets with the same names involve different preprocessing, which leads to unfair comparison. We briefly review all the datasets below and specify the operations to perform before applying each dataset.

- **NYT** (Riedel et al., 2010). NYT is the very first version among all the NYT-related datasets. It is based on the articles in New York Times[12]. We use the sentences from it to conduct the pilot experiment in Table 1. However, 1) it contains duplicate samples, e.g., 1504 in the training set; 2) It only labels the last word of an entity, which will mislead the evaluation results.

- **NYT10-HRL.** & **NYT11-HRL.** These two datasets are based on NYT. The difference is that they both contain *complete* entity mentions. NYT10 (Riedel et al., 2010) is the original one. and NYT11 (Hoffmann et al., 2011) is a small version of NYT10 with 53,395 training samples and a manually labeled test set of 368 samples. We refer to them as NYT10-HRL and NYT11-HRL after preprocessed by HRL (Takanobu et al., 2019) where they removed 1) training relation not appearing in the testing and 2) "NA" sentences. These two steps are almost adopted by all the compared methods. To compare fairly, we use this version in evaluations.

- **NYT21**. We provide relabel version of the test set of NYT11-HRL. The test set of NYT11-HRL still have false negative problem. Most of the samples in the NYT11-HRL has only one relation. We manually added back the missing triples to the test set.

- **SKE2019/SKE21**[13]. SKE2019 is a dataset in Chinese published by Baidu. The reason we also adopt this dataset is that it is currently the largest dataset available for relation extraction. There are 194,747 sentences in the training set and 21,639 in the validation set. We manually labeled 1,150 sentences from the test set with 2,765 annotated triples, which we refer to as SKE21. No preprocessing for this dataset is needed. We provide this data for future research[14].

5.2 Compared Methods and Metrics

We evaluate our model by comparing with several models on the same datasets, which are SOTA graphical model MultiR (Hoffmann et al., 2011), joint models SPTree (Miwa and Bansal, 2016) and NovelTagging (Zheng et al., 2017), recent strong SOTA models CopyR (Zeng et al., 2018b), HRL (Takanobu et al., 2019), CasRel (Wei et al., 2020), TPLinker (Wang et al., 2020). We also provide the result of automatically aligning Wikidata/CN-KBpedia with the corpus, namely *Match*, as a baseline. To note, we only keep the intersected relations, otherwise it will result in low precision due to the false negative in the original dataset. We report standard micro Precision (Prec.), Recall (Rec.) and F1 score for all the experiments. Following the previous works (Takanobu et al., 2019; Wei et al., 2020), we adopt partial match on these data sets for fair comparison. We also provide the results of exact match results of the methods we implemented, and only exact match on SKE2019.

5.3 Overall Comparison

We show the overall comparison result in Table 3. First, we observe that RERE consistently outperforms all the compared models. We find an interesting result that by purely aligning the database with the corpus, it already achieves surprisingly good overall result (surpassing MultiR) and relatively high precision (comparable to CoType in NYT11-HRL). However, the recall is quite low, which is consistent with our discussion in Section 2.1 that distant supervision leads to many false negatives. We also provide an ablation result where BERT is replaced with a bidirectional

[10] We do not use WebNLG (Gardent et al., 2017) and ACE04[11] because these datasets are not automatically labeled by distant supervision. WebNLG is constructed by natural language generation with triples. ACE04 is manually labeled.

[12] https://www.nytimes.com/

[13] http://ai.baidu.com/broad/download?dataset=sked

[14] download url.

	NYT10-HRL			NYT11-HRL			NYT21			SKE21		
	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1
KB Match	38.10	32.38	34.97	47.92	31.08	37.7	47.92	29.56	36.57	69.12	28.1	39.96
MultiR (Hoffmann et al., 2011)	-	-	-	32.8	30.6	31.7	-	-	-	-	-	-
SPTree (Miwa and Bansal, 2016)	49.2	55.7	52.2	52.2	54.1	53.1	-	-	-	-	-	-
NovelTagging (Zheng et al., 2017)	59.3	38.1	46.4	46.9	48.9	47.9	-	-	-	-	-	-
CoType (Ren et al., 2017)	-	-	-	48.6	38.6	43.0	-	-	-	-	-	-
CopyR (Zeng et al., 2018b)	56.9	45.2	50.4	34.7	53.4	42.1	-	-	-	-	-	-
HRL (Takanobu et al., 2019)	71.4	58.6	64.4	53.8	53.8	53.8	-	-	-	-	-	-
TPLinker (Wang et al., 2020)*	**81.19**	65.41	72.45	56.2	55.14	55.67	59.78	55.78	57.71	-	-	-
CasRel (Wei et al., 2020)*	77.7	68.8	73.0	50.1	58.4	53.9	58.64	56.62	57.61	-	-	-
RERE - LSTM	56.71	42.00	48.26	**56.46**	35.4	43.52	**62.06**	37.01	46.37	-	-	-
RERE	75.45	**72.50**	**73.95**	53.12	**59.59**	**56.23**	57.69	**61.69**	**59.62**	**-**	**-**	**-**
TPLinker (Wang et al., 2020)*(exact)	80.34	65.11	71.93	55.43	55.12	55.28	58.96	55.78	57.33	83.86	84.77	84.32
CasRel (Wei et al., 2020)*(exact)	75.12	65.72	70.11	47.88	55.13	51.25	55.06	54.49	54.78	86.94	85.96	86.45
RERE (exact)	74.90	71.97	73.4	52.40	58.91	55.47	56.97	60.93	58.88	90.44	84.20	87.21

Table 3: The main evaluation results of different models on NYT10-HRL, NYT11-HRL, and two hand labeled test sets NYT21 and SKE21 on by the compared method on the datasets. The results with only one decimal are quoted from (Wei et al., 2020). The methods with * are based on our re-implementation. Best partial (exact) match results are marked **bold** (underlined).

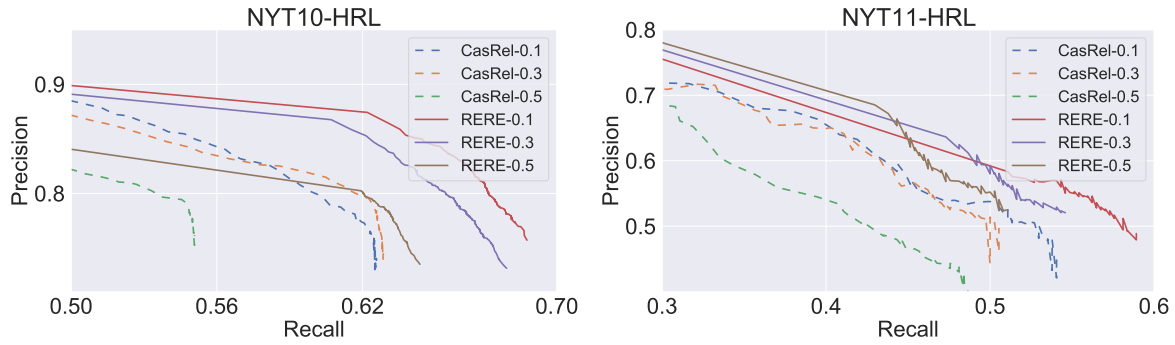

Figure 3: Precision-Recall Curve of RERE and CASREL under different false negative rates. Lines are better in the upper-right corner than the opposite. Note that the coordinates do not start from 0.

LSTM encoder (Graves et al., 2013) with randomly initialized weights. From the results we discover that even without BERT, our framework achieves competitive results against the previous approaches such as CoType and CopyR. This further prove the effectiveness of our RERE framework.

5.4 How Robust is RERE against False Negatives?

To further study how our model behaves when training data includes different quantity of false negatives, we conduct experiments on synthetic datasets. We construct five new training data by randomly removing triples with probability of 0.1, 0.3 and 0.5, simulating the situation of different FN rates. We show the precision-recall curves of our method in comparison with CASREL (Wei et al., 2020), the best performing competitor, in Figure 3. 1) The overall performance of RERE is superior to competitor models even when trained on a dataset with a 0.5 FN rate. 2) We show that the intervals of RERE between lines are smaller than CASREL, indicating that the performance decline under different FN rates of RERE is smaller. 3) The straight line before curves of our model means that there is no data point at the places where recall is very low. This means that our model is insensitive with the decision boundary and thus more robust.

6 Conclusion

In this paper, we revisit the negative data in relation extraction task. We first show that the false negative rate is largely underestimated by previous researches. We then systematically compare three

commonly adopted paradigms and prove that our paradigm suffers less from the overwhelming negative labels. Based on this advantage, we propose RERE, a pipelined framework that first detect relations at sentence level and then extract entities for each specific relation and provide a multi-label PU learning loss to recover false negatives. Empirical results show that RERE consistently outperforms the existing state-of-the-arts by a considerable gap, even when learned with large false negative rates.

Acknowledgments

This work is supported by National Key Research and Development Project (No. 2020AAA0109302), Shanghai Science and Technology Innovation Action Plan (No.19511120400) and Shanghai Municipal Science and Technology Major Project (No.2021SHZDZX0103). The authors would like to thank the anonymous reviewers for their constructive comments.

References

Jessa Bekker and Jesse Davis. 2018. Estimating the class prior in positive and unlabeled data through decision tree induction. In *Proceedings of AAAI*.

Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of SIGMOD*.

Yee Seng Chan and Dan Roth. 2011. Exploiting syntactico-semantic structures for relation extraction. In *Proceedings of ACL*, pages 551–560.

Danqi Chen. 2018. *Neural Reading Comprehension and Beyond*. Ph.D. thesis, Stanford University.

Md. Faisal Mahbub Chowdhury and A. Lavelli. 2012. Impact of less skewed distributions on efficiency and effectiveness of biomedical relation extraction. In *Proceedings of COLING*.

J. Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*.

Jun Feng, Minlie Huang, Li Zhao, Yang Yang, and Xiaoyan Zhu. 2018. Reinforcement learning for relation classification from noisy data. In *Proceedings of AAAI*, volume 32.

Claire Gardent, Anastasia Shimorina, Shashi Narayan, and Laura Perez-Beltrachini. 2017. Creating training corpora for NLG micro-planners. In *Proceedings of ACL*.

Matthew R Gormley, Mo Yu, and Mark Dredze. 2015. Improved relation extraction with feature-rich compositional embedding models. In *Proceedings of ACL*, pages 1774–1784.

Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton. 2013. Speech recognition with deep recurrent neural networks. In *2013 IEEE international conference on acoustics, speech and signal processing*, pages 6645–6649.

Pankaj Gupta, Hinrich Schütze, and Bernt Andrassy. 2016. Table filling multi-task recurrent neural network for joint entity and relation extraction. In *Proceedings of COLING*, pages 2537–2547.

R. 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 ACL*.

Wei Jia, Dai Dai, Xinyan Xiao, and Hua Wu. 2019. ARNOR: Attention regularization based noise reduction for distant supervision relation classification. In *Proceedings of ACL*, pages 1399–1408.

Arzoo Katiyar and Claire Cardie. 2017. Going out on a limb: Joint extraction of entity mentions and relations without dependency trees. In *Proceedings of ACL*, pages 917–928.

Q. Li and Heng Ji. 2014. Incremental joint extraction of entity mentions and relations. In *Proceedings of ACL*.

Xiaoya Li, Xiaofei Sun, Yuxian Meng, Junjun Liang, F. Wu, and J. Li. 2020. Dice loss for data-imbalanced nlp tasks. In *Proceedings of ACL*.

Xiaoya Li, Fan Yin, Zijun Sun, Xiayu Li, Arianna Yuan, Duo Chai, Mingxin Zhou, and J. Li. 2019. Entity-relation extraction as multi-turn question answering. In *Proceedings of ACL*.

Hongyu Lin, Yaojie Lu, Xianpei Han, and Le Sun. 2018. Adaptive scaling for sparse detection in information extraction. In *Proceedings of ACL*, pages 1033–1043.

Yankai Lin, Shiqi Shen, Zhiyuan Liu, Huanbo Luan, and Maosong Sun. 2016. Neural relation extraction with selective attention over instances. In *Proceedings of ACL*, pages 2124–2133.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Bonan Min, Ralph Grishman, Li Wan, Chang Wang, and David Gondek. 2013. Distant supervision for relation extraction with an incomplete knowledge base. In *Proceedings of HLT-NAACL*.

Mike Mintz, Steven Bills, Rion Snow, and Dan Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *Proceedings of ACL*.

Makoto Miwa and Mohit Bansal. 2016. End-to-end relation extraction using lstms on sequences and tree structures. In *Proceedings of ACL*, pages 1105–1116.

Makoto Miwa and Yutaka Sasaki. 2014. Modeling joint entity and relation extraction with table representation. In *Proceedings of EMNLP*, pages 1858–1869.

Marthinus Christoffel du Plessis, Gang Niu, and Masashi Sugiyama. 2015. Class-prior estimation for learning from positive and unlabeled data. *Machine Learning*, 106:463–492.

Xiang Ren, Zeqiu Wu, Wenqi He, Meng Qu, Clare R Voss, Heng Ji, Tarek F Abdelzaher, and Jiawei Han. 2017. Cotype: Joint extraction of typed entities and relations with knowledge bases. In *Proceedings of WWW*, pages 1015–1024.

S. Riedel, Limin Yao, and A. McCallum. 2010. Modeling relations and their mentions without labeled text. In *Proceedings of ECML/PKDD*.

Roland Roller, Eneko Agirre, Aitor Soroa, and Mark Stevenson. 2015. Improving distant supervision using inference learning. In *Proceedings of ACL*, pages 273–278.

Cicero dos Santos, Bing Xiang, and Bowen Zhou. 2015. Classifying relations by ranking with convolutional neural networks. In *Proceedings of ACL*, pages 626–634.

Ryuichi Takanobu, Tianyang Zhang, Jiexi Liu, and Minlie Huang. 2019. A hierarchical framework for relation extraction with reinforcement learning. In *Proceedings of AAAI*, volume 33, pages 7072–7079.

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 *Proceedings of NeuroIPS*, pages 6000–6010.

Denny Vrandecic and M. Krötzsch. 2014. Wikidata: a free collaborative knowledgebase. *Communications of the ACM*, 57:78–85.

Shuohang Wang and Jing Jiang. 2017. Machine comprehension using match-lstm and answer pointer. In *Proceedings of ICLR*.

Yucheng Wang, Bowen Yu, Yueyang Zhang, Tingwen Liu, Hongsong Zhu, and Limin Sun. 2020. TPLinker: Single-stage joint extraction of entities and relations through token pair linking. In *Proceedings of COLING*, pages 1572–1582.

Zhepei Wei, Jianlin Su, Yue Wang, Y. Tian, and Yi Chang. 2020. A novel cascade binary tagging framework for relational triple extraction. In *Proceedings of ACL*.

Chenhao Xie, Qiao Cheng, Jiaqing Liang, Lihan Chen, and Y. Xiao. 2020. Collective loss function for positive and unlabeled learning. *ArXiv*, abs/2005.03228.

Bo Xu, Yong Xu, Jiaqing Liang, Chenhao Xie, Bin Liang, Wanyun Cui, and Y. Xiao. 2017. CN-DBpedia: A never-ending chinese knowledge extraction system. In *Proceedings of IEA/AIE*.

Wei Xu, Raphael Hoffmann, Le Zhao, and Ralph Grishman. 2013. Filling knowledge base gaps for distant supervision of relation extraction. In *Proceedings of ACL*, pages 665–670.

Bowen Yu, Zhenyu Zhang, Xiaobo Shu, Tingwen Liu, Yubin Wang, Bin Wang, and Sujian Li. 2020. Joint extraction of entities and relations based on a novel decomposition strategy. In *Proceedings of ECAI*.

Xiaofeng Yu and Wai Lam. 2010. Jointly identifying entities and extracting relations in encyclopedia text via a graphical model approach. In *Proceedings of COLING*, pages 1399–1407.

Daojian Zeng, Kang Liu, Yubo Chen, and Jun Zhao. 2015. Distant supervision for relation extraction via piecewise convolutional neural networks. In *Proceedings of EMNLP*.

Daojian Zeng, Kang Liu, Siwei Lai, Guangyou Zhou, Jun Zhao, et al. 2014. Relation classification via convolutional deep neural network. In *Proceedings of COLING*, pages 2335–2344.

Xiangrong Zeng, Shizhu He, Kang Liu, and Jun Zhao. 2018a. Large scaled relation extraction with reinforcement learning. In *Proceedings of AAAI*, volume 32.

Xiangrong Zeng, Daojian Zeng, Shizhu He, Kang Liu, and Jun Zhao. 2018b. Extracting relational facts by an end-to-end neural model with copy mechanism. In *Proceedings of ACL*.

Tianyang Zhao, Zhao Yan, Y. Cao, and Zhoujun Li. 2020. Asking effective and diverse questions: A machine reading comprehension based framework for joint entity-relation extraction. In *Proceedings of IJCAI*.

Suncong Zheng, Feng Wang, Hongyun Bao, Yuexing Hao, Peng Zhou, and Bo Xu. 2017. Joint extraction of entities and relations based on a novel tagging scheme. In *Proceedings of ACL*, pages 1227–1236.