Front. Comput. Sci., 2025, 19(10): 1910352
https://doi.org/10.1007/s11704-024-3983-6

RESEARCH ARTICLE

E^2CNN: entity-type-enriched cascaded neural network for Chinese financial relation extraction

Mengfan LI[1], Xuanhua SHI (✉)[1], Chenqi QIAO[1], Xiao HUANG[2], Weihao WANG[1], Yao WAN[1], Teng ZHANG[1], Hai JIN[1]

1 National Engineering Research Center for Big Data Technology and System, Services Computing Technology and System Lab, Cluster and Grid Computing Lab, Huazhong University of Science and Technology, Wuhan 430074, China

2 Department of Computing, The Hong Kong Polytechnic University, Hong Kong 999077, China

Abstract Knowledge Graphs (KGs) are pivotal for effectively organizing and managing structured information across various applications. Financial KGs have been successfully employed in advancing applications such as audit, anti-fraud, and anti-money laundering. Despite their success, the construction of Chinese financial KGs has seen limited research due to the complex semantics. A significant challenge is the overlap triples problem, where entities feature in multiple relations within a sentence, hampering extraction accuracy – more than 39% of the triples in Chinese datasets exhibit the overlap triples. To address this, we propose the Entity-type-Enriched Cascaded Neural Network (E^2CNN), leveraging special tokens for entity boundaries and types. E^2CNN ensures consistency in entity types and excludes specific relations, mitigating overlap triple problems and enhancing relation extraction. Besides, we introduce the available Chinese financial dataset FinCorpus.CN, annotated from annual reports of 2,000 companies, containing 48,389 entities and 23,368 triples. Experimental results on the DUIE dataset and FinCorpus.CN underscore E^2CNN's superiority over state-of-the-art models.

Keywords financial knowledge graph, overlap triples, cascaded neural network, relation extraction

1 Introduction

Knowledge graphs have become a widely adopted standard for knowledge representation in the semantic web, where knowledge is encoded as a collection of "facts". Typically, the facts are expressed as triples of the form (*subject, predicate, object*), where the subject and object are entities, and the predicate indicates a relation between them. In general, knowledge graphs can be applied to support various downstream tasks, such as question answering [1,2], data integration [3], and fact-checking [4].

Currently, significant efforts have been invested in constructing a knowledge graph (KG) with a primary focus on named entity recognition [5,6] and relation extraction [7–9]. Several domain knowledge graphs have already been established, including those for food [10] and manufacturing [11]. In this research, we focus on constructing a Chinese financial knowledge graph. This resource holds potential applications in diverse anti-fraud services, such as consumer credit services [12], credit card applications [13], fraud detection, and identification [14]. Although some attempts have been made to build language-independent neural networks for knowledge graph construction from diverse language corpora [15], extracting relations from the Chinese financial corpus remains challenging due to its unique characteristics.

One obstacle is the presence of the same entity appearing in multiple relations within a single sentence. According to our statistics from the collected Chinese financial corpus, nearly 62.91% of entities exhibit this phenomenon, which is also known as the overlap triple problem [16,17]. In Fig. 1, we provide an example to illustrate the entity pair overlap (EPO) and single entity overlap (SEO) patterns of overlap triples. For instance, consider the sentence "Suzhou Jiangnan Film Vice President Zhang Hua has become a board member and investor of Yunhai Film." Multiple relations exist between "Zhang Hua" and "Yunhai Film". Furthermore, "Suzhou Jiangnan Film" serves as the subject in the triple (Suzhou Jiangnan Film, locate, Suzhou) and as the object in the triple (Zhang Hua, belong, Suzhou Jiangnan Film).

To tackle these issues, we employ a cascaded neural network. Initially, we detect a group of candidate subjects. Thereafter, for each subject, we proceed to extract all related objects via pre-defined relations, addressing the entity pair overlap. For single entity overlap, since extraction is structured for each relation, every subject can match its potential objects in different relations.

Besides, we discover that the relation types in the Chinese financial corpus exhibit a strong correlation with the entity types, and both the boundaries and types of entities offer a valuable opportunity. Such improvement materializes in a two-fold manner. **Consistency and exclusion in relation**

Received December 5, 2023; accepted August 26, 2024

E-mail: xhshi@hust.edu.cn

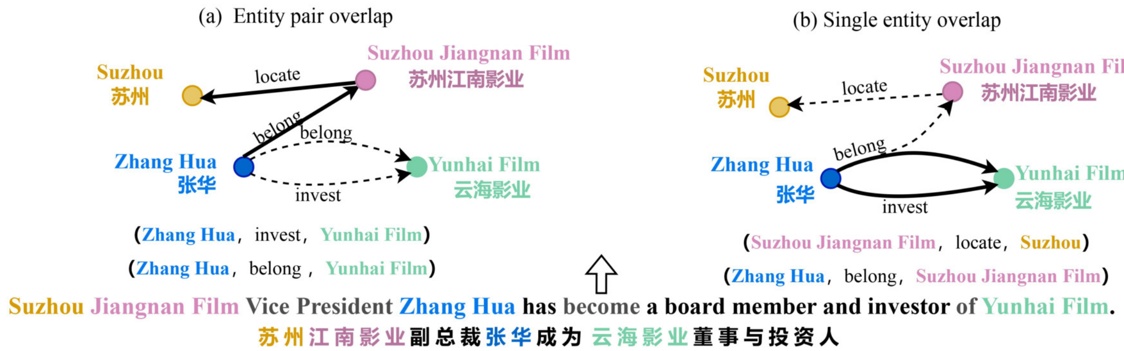

Fig. 1 Illustrative instances of two pattern overlaps in triples: (a) Entity pair overlap occurs when identical subjects and objects are involved in distinct relations, such as "Zhang Hua" and "Yunhai Film" simultaneously participate in "invest" and "belong"relations. (b) Single entity overlap arises when the same entity is present in different triples, as observed with "Suzhou Jiangnan Film", which features in three two distinct triples within a single sentence

type. Entities under the same subject and pre-defined relation embody a common type. Besides, for a specific entity and relation, coexistence is not present. As an illustration in Fig. 2, entities like " Suzhou Jiangnan Film" and "Yunhai Film" are typified as the "company" type. Both are identified under the "belong" relation. The head entity, such as "Zhang Hua", belongs to the "person" type, and likely matching relations for objects include "belong", "cooperate", and "invest" (with a "Company" type object), as opposed to relations like "launch", "own", "locate", and "apply to". This exclusion helps prevent the occurrence of false relations. **Defining entity boundaries.** On defining the entity type, a coincidental finding often is the demarcation of the entity, delineating the beginning and end of the entity. Clearly defined boundaries can aid in classifying nested entities. For example, in the triple ("Suzhou Jiangnan Film", locate, "Suzhou"), through the entity boundary, we can confirm the presence of two distinct entities within the phrase "Suzhou Jiangnan Film".

In the light of the above, we propose an Entity-type-Enriched Cascaded Neural Network (E^{2}CNN) for constructing financial knowledge graphs. Specifically, we utilize span-type identification, which involves employing special tokens [e:type] and [/e:type], to demarcate the boundaries and types of extracted entities. Following this, relying on the entity-type enriched representation, we engage the cascaded neural network to extract all subjects and then extract their corresponding objects for each relation type, simultaneously. The primary contributions of this paper are as follows:

- We propose a novel entity-type-enriched cascaded neural network (E^{2}CNN) that considers the overlap triple problem and entity-type information to construct a Chinese financial knowledge graph. We investigate the challenge of relation extraction from specific financial application scenarios and data characteristics. The learned knowledge graph is publicly available on GitHub[1).
- We release a Chinese financial dataset (termed FinCorpus.CN) based on a collection of Chinese financial company annual reports. The data annotation process involves manual annotation and cross-validation, resulting in 7 financial relation types and 6 financial entity types from 2,000 companies.
- We conduct comprehensive experiments on the publicly available dataset DUIE and our newly established FinCorpus.CN. The results demonstrate that E^{2}CNN outperforms several state-of-the-art models, highlighting its effectiveness in relation extraction.

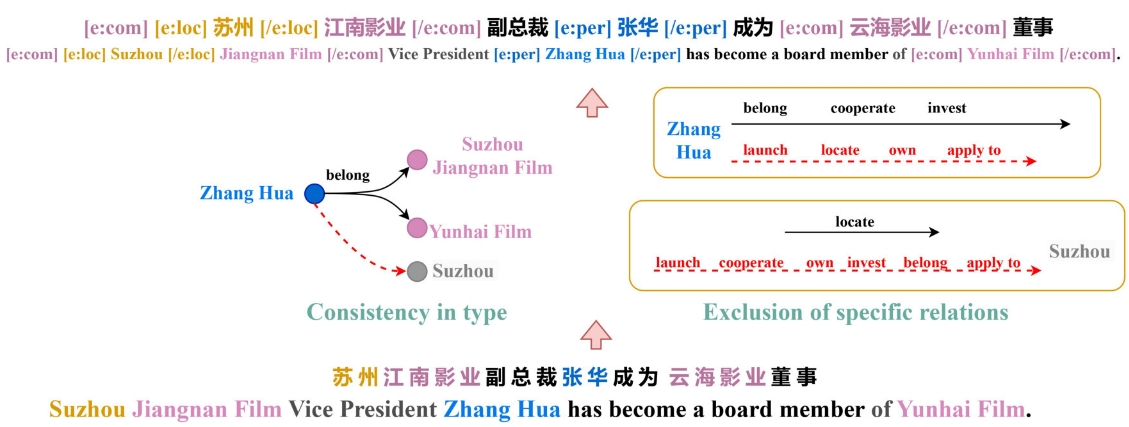

Fig. 2 Instances demonstrate how type information provides an opportunity for precise extraction from both the *Consistency in Type* and *Exclusion of Specific Relations* aspects. The red dotted lines highlight potential incorrect or unmatched relations, while black solid lines depict possible relations. Type labels and entity boundaries are added at the beginning and end of the identified entities, enriching their representations

1) See github.com/CGCL-codes/E-2CNN website.

2 Methodology

Consider the Chinese financial corpus, i.e., $C = \{s_1, s_2, \ldots, s_m\}$ (m is the corpus's total number of sentences). The ith sentence consists of multiple words, i.e., $s_i = \{w_1, w_2, \ldots, w_n\}$ (n is the number of words contained in the ith sentence). Our goal is to extract relation facts and construct a knowledge graph $G = \langle V, E\rangle$ based on the relation facts, where the subject and object are vertices and the predicate is an edge between those vertices. Figure 3 shows an overview of the proposed E^{2}CNN. It contains two principal components: span-type identification and cascaded neural network. Essential notations and definitions we used in this paper can be found in Table 1.

Table 1 Main symbols and definitions in this paper

Symbol	Description
s	Specific sentence in the corpus
T	Predefined entity type set
P	Entity type probability threshold
D_{entity}	Predicted entity set
$p:q$	Candidate span
$E_{p:q}$	Encoding vector of candidate span
$P_{p:q}$	Probability about $p:q$ is predicted as each type
t_s	Predicted entity type
$E_{[e:type]}, E_{[/e:type]}$	Beginning and end type label embedding of the entity
$E_s^{'}$	Enriched entity label embedding of sentence s
$P_{beg_{sub}:j}, P_{beg_{obj}:j}$	Probability of position j predicted as the beginning of a subject/ object
$P_{end_{sub}:j}, P_{end_{obj}:j}$	Probability of position j predicted as the end of a subject/ object
$W_{beg:sub}, b_{beg:sub}$	Adjustable parameters about
$W_{beg:obj}, b_{beg:obj}$	Boundary prediction
$\mathcal{L}_e$	Span-type identification cross-entropy loss
$\mathcal{L}_{sub}, \mathcal{L}_{obj}$	Final loss of the subject/ joint object extraction

2.1 Span-type-identification

Span refers to a collection of consecutive words in the sequence, the span-based method learns a deep representation for each possible span, classifying it to its corresponding type. The process is described in Algorithm 1. $p:q$ is the candidate span consisting of consecutive words with subscripts from p to q in the sentence s, $E_{p:q}$ is the encoding vector of candidate span $p:q$, $P_{p:q}$ is the probability that $p:q$ is predicted as each type.

Specifically, we first vectorize the input sentences s using the pre-trained model BERT to get the contextualized representation $\boldsymbol{E}_s \in \mathbb{R}^{n\times d}$ for all tokens in a sentence [18], and then enumerates all possible spans in the input sentence. Its

Algorithm 1 Process of span type identification

Require: Input sentence s, pre-trained model BERT, entity type probability threshold P, pre-defined entity types T.
Ensure: Predicted entity set D_{entity}

```
E_s = BERT(s) = {e_1, e_2, ..., e_n}
D_entity = ∅
for p = 1 → n − L do
    for q = p → p + L do
        p : q = [w_p, w_p + 1, ..., w_q]
        E_p:q = emb(p : q)
        P_p:q = softmax(W_e * E_p:q + b_e)
        if max(P_p:q) ⩾ P then
            mark corresponding entity type as t_s
            D_entity = D_entity ∪ (p, q, t_s)
        end if
    end for
end for
return D_entity
```

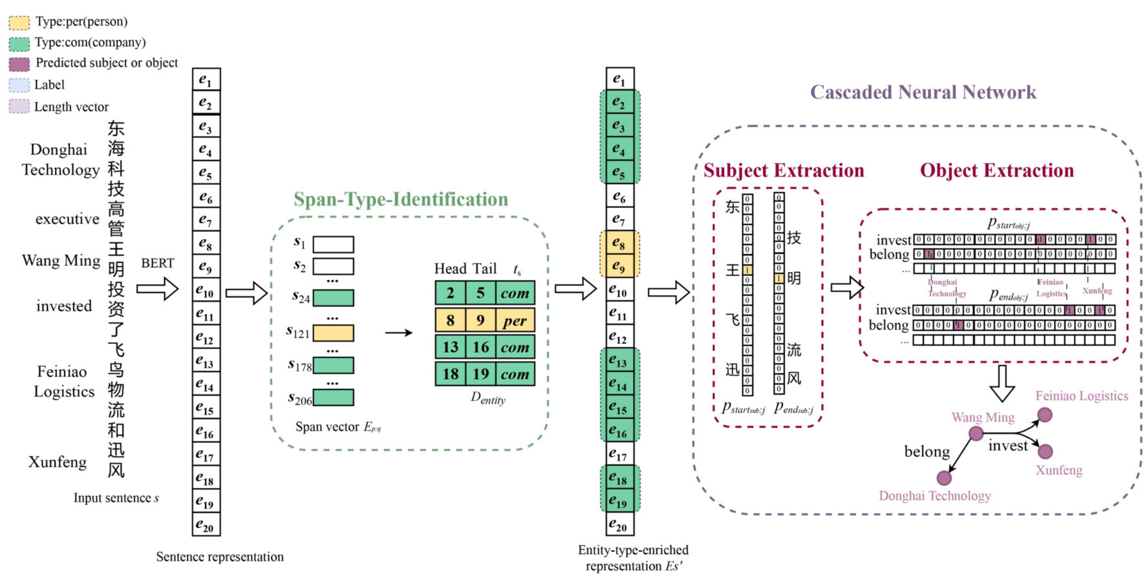

Fig. 3 An overview of our E^{2}CNN model. In this example, given an input sentence, our relation extraction is expected to extract that "Donghai Technology", "Feiniao Logistics" and " Xunfeng" are entities of type "company", "Wang Ming" is "person" type, as well as ("Wang Ming", *belong*, "Donghai Technology") and ("Wang Ming", *invest*, {"Feiniao Logistics", "Xunfeng"}). (1) The Span-Type-Identification module identifies the entities and corresponding types and obtains entity-enriched representations by incorporating the predicted types' information into the original corpus. (2) In the cascaded neural network module, the subject extraction module predicts both the beginning and end position of the subject, leveraging type-enriched information, object extraction detects all objects corresponding to the subjects obtained by the subject extraction under each predefined relation

original vector is represented as $E_{p:q}=\{e_p,e_{p+1},\ldots,e_q\}(1\leqslant p\leqslant q\leqslant n)$, where e_p and e_q denote the vector representation of head and tail of a span, respectively.

Based on the initial embedding, general span-based methods have two mainstream ways for span type identification. One concats the hidden states of the first and last tokens and combines these with a length feature vector to create the span representation, which is then classified [19–21]. The process is as follows:

$$\boldsymbol{E}_{p:q}=[e_p;e_q;l_{q-p+1}], \tag{1}$$

where $l_{q-p+1}\in\mathbb{R}^{d_l}$ represents the length feature corresponding to the length of the span vector, d_l represents the dimension of the length feature. The other is using a multi-head Biaffine decoder [22,23] to get the scores matrix for each enumerated span:

$$\boldsymbol{E}_p=LeakyReLU(\boldsymbol{E}_sW_p), \tag{2}$$

$$\boldsymbol{E}_q=LeakyReLU(\boldsymbol{E}_sW_q), \tag{3}$$

$$\boldsymbol{E}_{p:q}=MHBiaffine(E_p,E_q), \tag{4}$$

where $w_p,W_q\in\mathbb{R}^{d\times h}$ and h is the hidden size. *MHBiaffine* is the multi-head Biaffine decoder[2)]. Therefore, the final representation of span can be denoted as

$$E_{p:q}=emb(p:q),$$

where the *emb* method can adopt Eqs. (1) and (2). The probability of the span for each entity type is calculated as follows:

$$P_{p:q}=\mathrm{softmax}(W_e*E_{p:q}+b_e),$$

where W_e and b_e are the learnable parameters of the fully connected layer. Since we have pre-defined 6 entity types, $P_{p:q}\in\mathbb{R}^{6\times1}$. Besides, we prune out the non-entity spans (none of its type probability is above the predefined probability threshold P), and then sort the remained spans based on their maximum entity score $P_{p:q}$, mark the corresponding type t_s as the predicted entity type. Finally, the predicted entity set D_{entity} is obtained as follows:

$$D_{entity}=\{(p_1,q_1,t_1),\ldots,(p_z,q_z,t_z)\},$$

where z represents the length of the set of predicted entities, and each predicted entity consists of three elements, where p_z represents the beginning position of the zth entity, q_z represents the z-th entity the end position of the entity, and t_z represents the predicted entity type of the zth entity.

2.2 Entity-type-enriched representation

We use two special token $[e:type]$ and $[/e:type]$ to mark the extracted entities, which indicate the beginning and end of the entity, respectively. Taking Fig. 3 as an example, “Donghai Technology” is the “company type”, and “Wang Ming” is the “person” type. Thus, the embedding of token “Dong” and “Wang” are added with the corresponding beginning label embeddings “$E_{[e:com]}$”, “$E_{[e:per]}$”, and the embedding of “Technology” and “Ming” are added with the end label embeddings “$E_{[/e:com]}$”, “$E_{[/e:per]}$”. The enriched embedding with the entity label is formulated as

$$E_s^{'}=\{\ldots,E_{[e:type]}+e_p,\ldots,E_{[/e:type]}+e_q,\ldots\}.$$

For each span in the D_{entity}, the beginning label embedding is added to the head of span e_p, and the end label embedding is added to its tail position e_q. Likewise, the representation containing the entity type labels is obtained through the BERT [18].

Note: We primarily concentrate on identifying the head and tail positions. Enriching entity types is to strengthen the boundary and type information of the entity. Thus, we primarily concentrate on adding special token embeddings, $E_{[e:type]}$ and $E_{[/e:type]}$, on the head and tail positions of the entity as opposed to incorporating them throughout the entire entity, which prevents disruptions to the sequential operations.

2.3 Cascaded neural network

In the cascaded neural network module, we extract all subjects first, and then extract their corresponding objects for each relation type.

2.3.1 Subject extraction module

Subject extraction aims to discern the subjects within the entities, focusing on semantics. The process employs a distinct binary classifier to predict whether each position in the sequence is the beginning or end of the subject. For each position e_j in $E_s^{'}$, we use a sigmoid function to obtain the probability that the position is the beginning position of the subject, $p_{beg_{sub}:j}$, and the probability of the end position, $p_{end_{sub}:j}$, respectively.

$$p_{beg_{sub}:j}=sigmoid(W_{beg:sub}*e_j+b_{beg:sub}), \tag{5}$$

$$p_{end_{sub}:j}=sigmoid(W_{end:sub}*e_j+b_{end:sub}), \tag{6}$$

where $W_{beg:sub}$, $b_{beg:sub}$, $W_{end:sub}$, and $b_{end:sub}$ are the learnable parameters of the subject classifiers. The calculated probability of this position is compared to the predetermined probability threshold h_{bar}, and if it is higher, it’s counted as a boundary for the subject and marked as 1; if not, it is marked as 0. The predicted subject set is then compiled by pairing the 1 markers at the beginning and the end position. Then the final subject representation of the m-th predicted subject $e^{'}_{sub_m}$ is calculated as follows:

$$e^{'}_{sub_m}=\frac{1}{2}(e^{'}_{beg:m}+e^{'}_{end:m}), \tag{7}$$

which is the average of the beginning and end tokens of the entity embeddings.

2.3.2 Relation-specific objects extraction

Assuming that for mth subject in the candidate subjects, the input for the relation-specific objects extraction is as follows:

2) See github.com/yhcc/CNN_Nested_NER website.

$$E_m = E_s^{'} \oplus e_{sub_m}^{'}, \tag{8}$$

where $\oplus$ represents the matrix addition operation. For any word vector $e_j \in E_m$, any relation type r in the relation type set R, we use the sigmoid function to convert the output of the fully connected layer into probabilities.

$$P_{beg_{obj}:j} = Sigmoid(W_{beg:obj} * e_j + b_{beg:obj}),$$

$$P_{end_{obj}:j} = Sigmoid(W_{end:obj} * e_j + b_{end:obj}),$$

where $W_{beg:obj}$ and $b_{beg:obj}$ represent the adjustable parameters of the object beginning point, while $W_{end:obj}$ and $b_{end:obj}$ denote the end point. The predicted probabilities are symbolized as $P_{beg_{obj}:j}$ and $P_{end_{obj}:j}$. If the probability exceeds the threshold t_{bar}, the marker bit at this position is set as 1; if not, it is set as 0. Ultimately, the beginning and the end positions marked with 1 are paired to identify all candidate objects obj_{pred}.

Discussion: How does the cascaded neural network address the identified overlap issues?

For each subject, we extract all the corresponding objects based on every pre-defined relation. Consequently, for the same entity pair, under different relation classifiers, we can identify probable objects. Choosing a given subject, according to the relations, to extract objects, rather than identifying which relation the entity pair belongs to, can aid in mining multiple relations or objects.

For the extra subject-object overlap (SOO) problem, which means there are nested entities in the subject or object, such as the triple ("Suzhou Jiangnan Film", locate, "Suzhou"), span-type-identification can enumerate every candidate span set, so both "Suzhou" and "Suzhou Jiangnan Film" can be identified as entities. Besides, the cascaded neural network can iterate each position, in turn, to judge the entity boundary, so "zhou" and "Film" can be assuredly distinguished.

2.4 Training

For the span-type identification task, we tune the pre-trained language model using task-specific cross-entropy loss as follows:

$$\mathcal{L}_e = -\sum_{s \in span} \log P_s(t_s|s), \tag{9}$$

where t_s is the annotated entity type.

$$\mathcal{L}_{sub:beg} = -\sum_{k=1}^{n} x_{k:beg} \log p_{k:beg} + (1 - x_{k:beg}) \log(1 - p_{k:beg}),$$

$$\mathcal{L}_{sub:end} = -\sum_{k=1}^{n} x_{k:end} \log p_{k:end} + (1 - x_{k:end}) \log(1 - p_{k:end}),$$

where n represents the length of the sentence, and $x_{k:beg}$ and $x_{k:end}$ represent the true label of the kth word for the beginning and end positions of the subject, respectively. If the kth word is the subject beginning position, then $x_{k:beg}$ is 1; otherwise, it is 0. $p_{k:beg}$ and $p_{k:end}$ represent the probability of the kth word being predicted as the subject boundary position. The final loss function of the subject extraction process is the sum of the loss functions, given by

$$\mathcal{L}_{sub} = \mathcal{L}_{sub:beg} + \mathcal{L}_{sub:end}.$$

In the cascaded neural network, for a given relation r in the predefined relation types R, the loss function of the module that predicts the beginning position of the object is as follows:

$$\mathcal{L}_{r:beg} = -\sum_{j=1}^{n} y_{j:beg} \log p_{j:beg} + (1 - y_{j:beg}) \log(1 - p_{j:beg}).$$

Correspondingly, the loss function of the module that predicts the end position of the object is

$$\mathcal{L}_{r:end} = -\sum_{j=1}^{n} y_{j:end} \log p_{j:end} + (1 - y_{j:end}) \log(1 - p_{j:end}),$$

where $y_{j:beg}$ and $y_{j:end}$ represent the true label of the jth word for the beginning and end positions of the object, respectively. $p_{j:beg}$ and $p_{j:end}$ represent the probability of the jth word being predicted as the object boundary position. The final loss function of the joint extraction process is the sum of the loss functions for each relation type. The formula is as follows:

$$\mathcal{L}_{obj} = \sum_{r \in R} \mathcal{L}_{r:beg} + \mathcal{L}_{r:end}. \tag{10}$$

3 Experiments

In this section, we present the experimental setup for the E^{2}CNN model in relation extraction task.

3.1 Datasets

We employ two datasets: the large-scale Chinese general dataset DUIE [24] and our FinCorpus.CN datasets. The statistical information is presented in Table 2.

DuIE, developed by Baidu, is a large-scale Chinese dataset containing 450,000 instances and 49 common relation types. Due to the utilization of distant supervision for automatically generating labeled data, the dataset's quality is somewhat compromised, and the data distribution is not uniform. To address this, we clean and screen the original dataset, ultimately selecting 17 relation types and 15 entity types.

Table 2 The division of the dataset, the statistics of entity and relation types, and the number of sentences containing normal, single entity overlap, and entity pair overlap triples. In the public dataset DuIE, the proportion of overlap triples is 39.82%, while in our FinCorpus.CN dataset, the proportion of overlap triples reaches 56.91%

Dataset	Train	Dev	Test	Entity type	Relation type	Normal	Single entity overlap	Entity pair overlap
DuIE	55,959	11,191	13,417	15	17	48,484 (60.18%)	29,786 (36.97%)	2,297 (2.85%)
FinCorpus.CN	4,956	1,723	1,640	6	7	3,585 (43.09%)	4,508 (54.19%)	226 (2.72%)

FinCorpus.CN is a Chinese financial dataset constructed from a collection of annual reports published by 2,000 financial companies. Here, we use *brat*[3)], a widely used tool for entity and relation labeling, to manually annotate and cross-validate the data. The ontology diagram of FinCorpus.CN is shown in Fig. 4.

FinCorpus.CN comprises 8,319 samples, encompassing a total of 7 types of financial relations and 6 types of financial entities. Details regarding the involved entities and relations are elucidated in Table 3. The distribution of the number of entities and relation types in the samples is illustrated in Fig. 5.

Table 3 Description of entity and relation types in FinCorpus.CN

Entity type	Description
Company	Company name, abbreviation
Person	Character name
Location	City-level and above does not specify specific address
Product	Various products produced by the company
Business	Business activities of the company
Industry	Industry to which the company belongs to
Relation type	**Description**
Launch	Company carries out business
Locate	Location of the company or person
Cooperate	Cooperate between companies and individuals
Invest	Investment by individuals or enterprises in a company
Belong	Relation between the company, its industry, and individuals belonging to a specific company
Apply to	Product application in a specific industry
Own	Products owned by the company; parent company owning subsidiaries

3.2 Evaluation metrics

We employ precision (P), recall (R), and F1-measure (F1) as performance evaluation metrics. Precision represents the proportion of the number of true positive examples predicted by the model among the total positive examples it predicts. Recall, on the other hand, indicates the proportion of true positive examples predicted relative to the total number of true positive examples. The F1 value, being the harmonic average of precision and recall, serves as a judgment of the overall effectiveness of the model. Correct relation extraction entails the accurate prediction of entity boundaries of head and tail entities and the correct prediction of relation types.

3.3 Relation extraction test results and analysis

The F1 values of the relation extraction on DuIE and FinCorpus.CN reach 84.93% and 75.24%, respectively, which both exceeded the baseline models. E^{2}CNN is compared against five high-performing models:

- **Digie++** [25] uses BERT to obtain the token representations and T-Concat span representation in the input sentence, and then iteratively propagates the core inference and confidence of relation type through the span graph to refine the representation.
- **CasRel** [16] models relations as functions that map subjects to objects instead of discrete labels to better

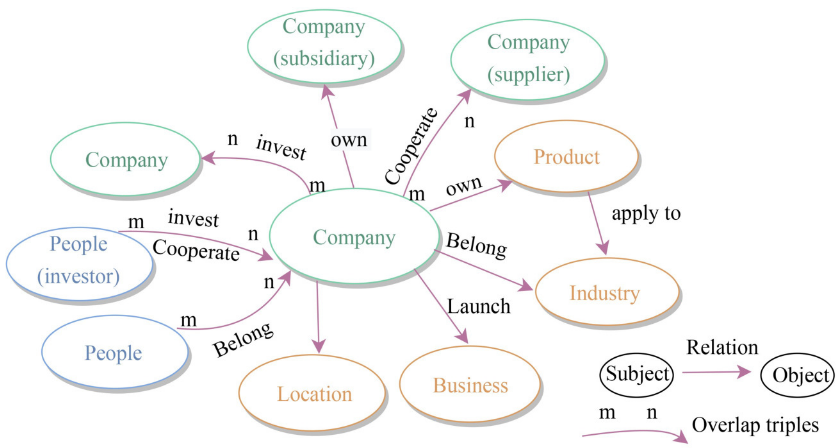

Fig. 4 Ontology diagram of FinCorpus.CN dataset

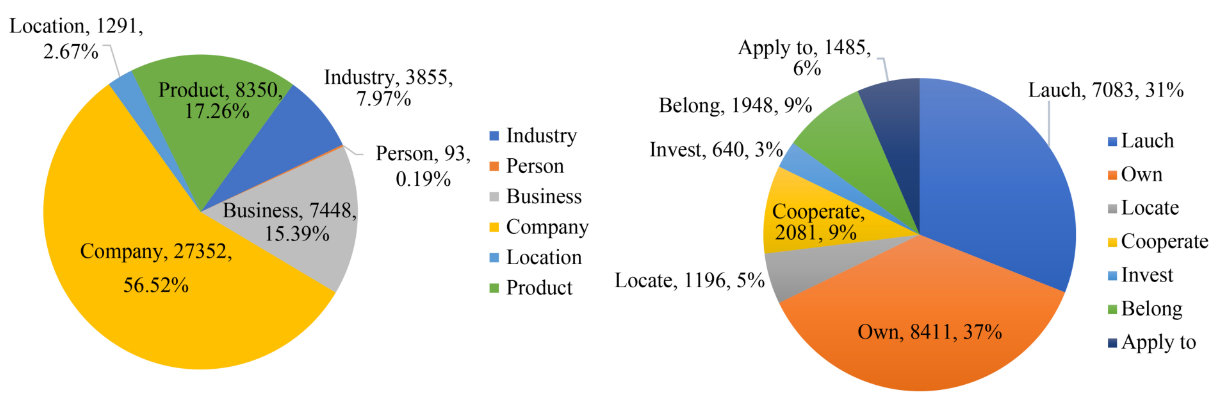

Fig. 5 Statistics about entities and relations in FinCorpus.CN. Left: Analysis of the number and proportion of entity of FinCorpus.CN; Right: analysis of the number and proportion of relation of FinCorpus.CN

handle the overlap problem. However, it does not consider the effect of entity type on relation extraction.

- **PL-Marker** [26] summarizes the existing work on span representation and adopts a fusion subject-oriented packing scheme in the span pair model. For the subject span, entity markers are applied by inserting [S] and [/S] before and after the span to indicate the beginning and end of the span, respectively.
- **BIRTE** [27] proposes a paralleled bidirectional extraction framework to extract all possible subject–object pairs and assigns all possible relations for pairs using a biaffine model. Additionally, it introduces a share-aware mechanism to address the issue of convergence rate inconsistency.
- **OD-RTE** [28] approaches the relation extraction task by treating it as an object detection problem. Additionally, it introduces the bidirectional diagonal walk decoding algorithm for extracting all types of triples.

Table 4 shows intuitively that the performance of our E^{2}CNN model is significantly better than previous models in terms of end-to-end relation extraction. Our model outperforms the state-of-the-art (SOTA) model by 19.7% and 28.69% on the DUIE and FinCorpus.CN, respectively, on the F1 metric. Moreover, the experimental results reveal significant variations in the performance of the same model across the FinCorpus.CN and DuIE datasets. This discrepancy is primarily attributed to the dissimilarities between the datasets. The sentence structure in DUIE is relatively straightforward and consistent, making it easier to learn and extract information. However, the Chinese financial dataset FinCorpus.CN contains a substantial number of overlap triples, reflecting distinctive data features and syntactic complexity. Comparison of the statistics and sample cases from the DUIE and FinCorpus.CN datasets are presented in Fig. 6.

Table 4 Comparison of experimental results of relation extraction on DuIE and FinCorpus.CN

Datasets	**DuIE**			**FinCorpus.CN**		
Metrics	P/%	R/%	F1/%	P/%	R/%	F1/%
CasRel [16]	79.15	73.14	76.03	56.69	50.96	53.67
Dygie++ [25]	77.97	56.13	65.27	53.62	37.45	44.09
PL-Marker [26]	79.48	59.67	68.17	52.21	44.46	48.02
BIRTE [27]	79.77	68.03	73.43	49.59	53.98	51.69
OD-TRE [28]	79.01	69.92	74.19	53.12	54.10	53.61
E^{2}CNN **(ours)**	**96.16**	**95.31**	**95.73**	**86.59**	**78.53**	**82.36**

Table 5 compares the F1, P, and R scores of the E^{2}CNN and CasRel models on the normal, single entity overlap, and entity pair overlap triples. The table shows that E^{2}CNN surpasses CasRel on all triples, especially on the normal dataset, where E^{2}CNN achieves a 29.19% higher F1 score than CasRel. Moreover, E^{2}CNN also outperforms CasRel by 21.86% and 31.1% on the single entity overlap and entity pair overlap triples, respectively. This demonstrates that the E^{2}CNN model is more capable of handling complex relation extraction tasks.

Implementation details For a fair comparison, we employ the chinese-roberta-wwm-ext model [18] as the pre-trained BERT model in all our experiments. In the span type identification task, the model is trained with a learning rate of 1×10^{-5} over 100 epochs, employing a batch size of 8. For the relation extraction task, we adjust the batch size to 4 and conduct 10 epochs. Additionally, to accommodate the task requirements, we set a maximum sentence length of 300. The model is implemented using the PyTorch [29] framework. During our experiments, the threshold for object detection is consistently set to 0.5 for both the head and tail. It is worth noting that this parameter has the potential for optimization to enhance model performance. However, we have opted to postpone its fine-tuning to future work.

4 Ablation study

We conduct an ablation experiment on the effect of entity type on different relations and calculate the type distribution of subject and object in the triples predicted by the test set. Table 6 demonstrates the experimental results, after removing the type information, the proportions of false type and non-entity both increase, especially the proportion of non-entity

Datasets	Amount of data	Data format regularity	Instances
DUIE	80,000 samples, 17 relation types	The format of the dataset is more uniform and regular,usually "Entity 1...is...Entity 2"	• "Listening to Li Ming: Moonlight on the Water Lily Pond is a book published by Harmony Media Press in 2006." • "Lion Patrol is a fashion police and crime TV series produced by Pearl City Broadcasting Co., Ltd."
FinCorpus.CN	8,319 samples, 7 relation types	The format is chaotic and complex, irregular	• "Coincidentally, in January this year, Lingnan Jinyue Holding Group, the largest shareholder of Huaxin Futures, listed its 51% stake for sale, with a reserve price of 168 million yuan." • "Cooperation between Silver Phoenix Auto and partners who produce components at an advanced global level, such as SkySteel, Lingde Power, and Huatai, continues to grow."

Fig. 6 Dataset differences in experimental results analysis of the same model under different datasets

Table 5 E^{2}CNN and CasRel models both adopt the cascaded binary tagging framework, which can achieve more efficient extraction performance on overlap triples than normal triples. Note that the possible single entity overlap and entity pair overlap can be in the same sentence

Triples	**Normal**			**Single entity overlap**			**Entity pair overlap**		
Metrics	P/%	R/%	F1/%	P/%	R/%	F1/%	P/%	R/%	F1/%
E^{2}CNN	84.29	77.67	80.84	84.47	83.42	83.95	88.34	79.68	83.79
CasRel [16]	40.63	33.49	51.65	62.02	61.95	62.09	58.67	66.17	52.69

Table 6 Type distribution of the subjects and objects of the triples predicted in the test set, as well as the performance of relation extraction. Among them, true type means that the predicted subject or object belongs to the correct type annotated, false type means that the predicted subject or object belongs to a different type, and non-entity means that the predicted subject or object is not in the annotated entity set, that is, an incorrect entity is predicted

Model	Relation type	Subject			Object			Relation		
		True type	False type	Non-entity	True type	False type	Non-entity	P/%	R/%	F1/%
E^2CNN (w/o type)	Launch	1033	0	33	633	37	396	57.69	54.76	56.19
	Own	1552	3	132	1013	36	638	51.81	64.98	57.65
	Locate	221	0	7	96	1	131	39.04	53.94	45.29
	Cooperate	370	0	18	294	3	91	64.95	71.19	67.92
	Invest	75	0	1	71	0	5	59.21	33.83	43.06
	Belong	408	1	6	158	60	197	35.42	53.07	42.49
	Apply to	124	1	6	98	3	30	62.60	56.55	59.42
E^2CNN	Launch	1398	6	3	1382	0	25	92.68	88.29	90.43
	Own	1774	0	46	1748	5	67	80.33	84.46	82.34
	Locate	257	4	3	264	0	0	90.91	94.49	92.66
	Cooperate	386	0	8	386	2	6	79.44	73.65	76.43
	Invest	112	0	2	107	0	7	39.47	37.82	38.63
	Belong	359	1	1	359	1	1	91.14	79.66	85.01
	Apply to	388	7	14	402	0	7	84.11	81.13	82.59

increases significantly. E^2CNN achieves higher F1 scores on most relations. Moreover, our model significantly reduces the proportion of false type and non-entity predictions. This demonstrates that entity type information can enhance the model's ability to extract more accurate and relevant triples from the corpus.

5 Related work

Currently, many knowledge graphs have been developed in different domains, such as RcpKG [30] and FabKG [11] for food and manufacturing, respectively. However, few works focus on the financial domain. Building a financial knowledge graph can facilitate answering complex financial domain queries [31]. Knowledge Graphs (KGs) construction by extracting data from structured or unstructured data sources, especially textual texts, is of great importance to support services like question answering [32,33], fact checking [34,35], and data integration [36,37]. Over the past few years, various knowledge graphs including DBpedia [38], YAGO [39], and NELL [40], have been developed for general-purposed domains.

In the financial field, Elhammadi et al. [41] proposed a pipeline knowledge extraction that uses conditional random field (CRF) to filter data and combines Semantic Role Labeling (SRL) and pattern-based information extraction to extract domain-targeted noun/verb-mediated relations in financial news domain. However, the Chinese financial sector lacks a sufficient amount of technologies for the systematical and automatic construction of KGs.

Relational triples extraction [28,42] from natural language corpora is a crucial stage in the construction of massive KGs. The primary study area at present is the relation extraction approach based on deep neural networks, and the effect of supervised learning [43] method is far better than that of semi-supervised [44] and unsupervised methods [45]. The two main categories of current methods for extracting relation triples are pipeline-based methods and joint-based methods.

5.1 Pipeline-based methods

Generally, the pipeline of KGs construction can be organized as two subtasks: 1) *named entity recognition* [5,6,46], which aims to recognize the financial entities from natural-language sentences; 2) *relation extraction* [7,8], which aims to link the financial entities via a relation.

Early works [47–50] follow a pipeline-based paradigm, namely training one model to extract entities (i.e., *entities recognition*) and another model to classify relations between them (i.e., *relation extraction*). Zeng et al. [51] proposed utilizing CNN to learn word-level features, further learn sentence-level features through convolution, and then classify the features into relations, demonstrating the feasibility of applying deep learning models to the relation extraction task. Zhou et al. [52] suggested improving feature capture by adding the Attention Mechanism on top of BiLSTM. Zhong and Chen [20] present a pipeline approach for representing all candidate spans and identifying the corresponding entity types. They propose the importance of fused entity information at the relation model's input layer. Based on their work, three commonly used span representation extraction methods (i.e., T-concat, Solid Marker, and Levitated Marker) are outlined by Ye et al. [26]. They investigate how various span representations affect relation extraction model performance.

5.2 Joint-based methods

The joint-based methods [53] combine entity extraction and relation extraction to accomplish the extraction in an end-to-end mode. Zheng et al. [54] proposed a joint extraction method that turns the two subtasks into a unified sequence labeling task with BiLSTM for coding and decoding, avoiding the entity redundancy problem caused by shared parameters. Bekoulis et al. [55] combined CRF for entity extraction with multi-head selection for relation classification to achieve joint extraction.

The foundation of conventional joint models is characteristics [56,57] which require labor-intensive human work. Recent research [58–60] has explored neural network-based techniques, offering cutting-edge performance while minimizing manual effort. Miwa and Bansal [61] proposed a neural network to obtain information about dependency tree substructure, but their model only achieved joint extraction by sharing parameters, not joint decoding. In contrast to earlier methods, Zheng et al. [54] used a tagging scheme to model the relation extraction problem instead of adhering to the task classification of entity extraction versus relationship extraction, treating the relation triples as a whole and directly extracting the information at the triple level.

Recently, external knowledge has been employed to complement traditional context information such as part-of-speech (POS) labels [62]. Similar approaches are used in the field of Chinese NER and RE. Li et al. [63] incorporate word-level information into character sequence inputs to avoid segmentation errors. Xuan et al. [64] incorporated graph information into the character representation.

Despite the extensive study of joint-based relation extraction methods, much of the current work has not considered the substantial amount of hyper-relational instances in the corpus. The relation types were categorized by Zeng et al. [60] as formal, single-entity overlap, and entity-pair overlap. They then attempted to use the sequence-to-sequence model to solve overlap relation issues. Fu et al. [65] suggested a graph convolutional network-based method for this problem. Shang et al. [66] utilized a scoring-based classifier and a relation-specific horns tagging strategy to address issues like cascaded errors and redundant information. Even after the initial successes [67], existing approaches continue to see relations as separate labels, which makes it challenging for overlap relation extraction.

6 Conclusion and future work

In our study, we investigate methods for overlap triple extraction from Chinese financial corpora and offer a dataset for financial knowledge graph construction. We utilize a cascaded neural network to extract all subjects first, and then extract their corresponding objects for each relation type, concurrently. Meanwhile, the extracted information about entity boundaries and types provides an opportunity for better relation extraction. Consequently, we propose an Entity-type-Enriched Cascaded Neural Network (E^{2}CNN) to enrich our cascaded neural network with entity type information for better relation extraction. We perform comprehensive experiments on the open-source Chinese datasets DUIE and our Chinese financial dataset (termed FinCorpus.CN). Experimental results demonstrate the effectiveness of E^{2}CNN when compared with several SOTA baselines and by using entity type information, E^{2}CNN enhances the relation extraction task performance.

Acknowledgements This work was supported in part by the National Key R&D Program of China (Grant No. 2020AAA0108501).

Competing interests The authors declare that they have no competing interests or financial conflicts to disclose.

References

1. Saxena A, Chakrabarti S, Talukdar P. Question answering over temporal knowledge graphs. In: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing. 2021, 6663−6676
2. Zhang M, He T T, Dong M. Meta-path reasoning of knowledge graph for commonsense question answering. Frontiers of Computer Science, 2024, 18(1): 181303
3. Collarana D, Galkin M, Traverso-Ribón I, Lange C, Vidal M E, Auer S. Semantic data integration for knowledge graph construction at query time. In: Proceedings of the 11th IEEE International Conference on Semantic Computing. 2017, 109−116
4. Kim J, Choi K S. Unsupervised fact checking by counter-weighted positive and negative evidential paths in a knowledge graph. In: Proceedings of the 28th International Conference on Computational Linguistics. 2020, 1677−1686
5. Sang E F T K, De Meulder F. Introduction to the CoNLL-2003 shared task: language-independent named entity recognition. In: Proceedings of the 7th Conference on Natural Language Learning at HLT-NAACL 2003. 2003, 142−147
6. Ratinov L, Roth D. Design challenges and misconceptions in named entity recognition. In: Proceedings of the 13th Conference on Computational Natural Language Learning. 2009, 147−155
7. Zelenko D, Aone C, Richardella A. Kernel methods for relation extraction. In: Proceedings of 2002 Conference on Empirical Methods in Natural Language Processing. 2002, 71−78
8. Bunescu R C, Mooney R J. A shortest path dependency kernel for relation extraction. In: Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing. 2005, 724−731
9. Zheng Z Y, Liu Y, Li D, Zhang X J. Distant supervised relation extraction based on residual attention. Frontiers of Computer Science, 2022, 16(6): 166336
10. Haussmann S, Seneviratne O, Chen Y, Ne'eman Y, Codella J, Chen C H, McGuinness D L, Zaki M J. FoodKG: a semantics-driven knowledge graph for food recommendation. In: Proceedings of the 18th International Semantic Web Conference. 2019, 146−162
11. Kumar A, Bharadwaj A G, Starly B, Lynch C. FabKG: a knowledge graph of manufacturing science domain utilizing structured and unconventional unstructured knowledge source. In: Proceedings of the Workshop on Structured and Unstructured Knowledge Integration (SUKI). 2022, 1−8
12. Kang Y Z, Jia N, Cui R B, Deng J. A graph-based semi-supervised reject inference framework considering imbalanced data distribution for consumer credit scoring. Applied Soft Computing, 2021, 105: 107259
13. Van Belle R, Mitrović S, De Weerdt J. Representation learning in graphs for credit card fraud detection. In: Proceedings of the 4th ECML PKDD Workshop on Mining Data for Financial Applications. 2020, 32−46
14. Zhan Q, Yin H. A loan application fraud detection method based on knowledge graph and neural network. In Proceedings of the 2nd International Conference on Innovation in Artificial Intelligence. 2018, 111−115
15. Iglesias-Molina A, Chaves-Fraga D, Priyatna F, Corcho O. Towards the

definition of a language-independent mapping template for knowledge graph creation. In: Proceedings of the 3rd International Workshop on Capturing Scientific Knowledge Co-located with the 10th International Conference on Knowledge Capture. 2019, 33−36
16. Wei Z, Su J, Wang Y, Tian Y, Chang Y. A novel cascade binary tagging framework for relational triple extraction. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. 2020, 1476−1488
17. Wang G, Zeng Y, Li R H, Qin H, Shi X, Xia Y, Shang X, Hong L. Temporal graph cube. IEEE Transactions on Knowledge and Data Engineering, 2023, 35(12): 13015−13030
18. Cui Y, Che W, Liu T, Qin B, Wang S, Hu G. Revisiting pre-trained models for Chinese natural language processing. In: Proceedings of Findings of the Association for Computational Linguistics: EMNLP 2020. 2020, 657−668
19. Xu W, Chen Y, Ouyang J. A streamlined span-based factorization method for few shot named entity recognition. In: Proceedings of 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation. 2024, 1673−1683
20. Zhong Z, Chen D. A frustratingly easy approach for entity and relation extraction. In: Proceedings of 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. 2021, 50−61
21. Dixit K, Al-Onaizan Y. Span-level model for relation extraction. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. 2019, 5308−5314
22. Yu J, Bohnet B, Poesio M. Named entity recognition as dependency parsing. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. 2020, 6470−6476
23. Yan H, Sun Y, Li X, Qiu X. An embarrassingly easy but strong baseline for nested named entity recognition. In: Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics. 2023, 1442−1452
24. Li S, He W, Shi Y, Jiang W, Liang H, Jiang Y, Zhang Y, Lyu Y, Zhu Y. DuIE: a large-scale Chinese dataset for information extraction. In: Proceedings of the 8th CCF International Conference on Natural Language Processing and Chinese Computing. 2019, 791−800
25. Wadden D, Wennberg U, Luan Y, Hajishirzi H. Entity, relation, and event extraction with contextualized span representations. In: Proceedings of 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing. 2019, 5784−5789
26. Ye D, Lin Y, Li P, Sun M. Packed levitated marker for entity and relation extraction. In: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics. 2022, 4904−4917
27. Ren F, Zhang L, Zhao X, Yin S, Liu S, Li B. A simple but effective bidirectional framework for relational triple extraction. In: Proceedings of the 15th ACM International Conference on Web Search and Data Mining. 2022, 824−832
28. Ning J, Yang Z, Sun Y, Wang Z, Lin H. OD-RTE: a one-stage object detection framework for relational triple extraction. In: Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics. 2023, 11120−11135
29. Dai H, Peng X, Shi X, He L, Xiong Q, Jin H. Reveal training performance mystery between TensorFlow and PyTorch in the single GPU environment. Science China Information Sciences, 2022, 65: 112103
30. Lei Z, Ul Haq A, Zeb A, Suzauddola M, Zhang D. Is the suggested food your desired? Multi-modal recipe recommendation with demand-based knowledge graph. Expert Systems with Applications, 2021, 186: 115708
31. Zehra S, Mohsin S F M, Wasi S, Jami S I, Siddiqui M S, Syed M K U R R. Financial knowledge graph based financial report query system. IEEE Access, 2021, 9: 69766−69782
32. Rony M R A H, Chaudhuri D, Usbeck R, Lehmann J. Tree-KGQA: an unsupervised approach for question answering over knowledge graphs. IEEE Access, 2022, 10: 50467−50478
33. Shang C, Wang G, Qi P, J H. Improving time sensitivity for question answering over temporal knowledge graphs. In: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics. 2022, 8017−8026
34. Lin P, Song Q, Wu Y. Fact checking in knowledge graphs with ontological subgraph patterns. Data Science and Engineering, 2018, 3(4): 341−358
35. Lin P, Song Q, Wu Y, Pi J. Discovering patterns for fact checking in knowledge graphs. Journal of Data and Information Quality, 2019, 11(3): 13
36. Cudré-Mauroux P. Leveraging knowledge graphs for big data integration: the XI pipeline. Semantic Web, 2020, 11(1): 13−17
37. Dandan R, Despres S. DIKG2: a semantic data integration approach for knowledge graphs generation from Web forms. In: Proceedings of the 34th International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems. 2021, 255−260
38. Lehmann J, Isele R, Jakob M, Jentzsch A, Kontokostas D, Mendes P N, Hellmann S, Morsey M, van Kleef P, Auer S, Bizer C. DBpedia – a large-scale, multilingual knowledge base extracted from Wikipedia. Semantic Web, 2015, 6(2): 167−195
39. Suchanek F M, Kasneci G, Weikum G. Yago: a core of semantic knowledge. In: Proceedings of the 16th International Conference on World Wide Web. 2007, 697−706
40. Mitchell T, Cohen W, Hruschka E, Talukdar P, Yang B, Betteridge J, Carlson A, Dalvi B, Gardner M, Kisiel B, Krishnamurthy J, Lao N, Mazaitis K, Mohamed T, Nakashole N, Platanios E, Ritter A, Samadi M, Settles B, Wang R, Wijaya D, Gupta A, Chen X, Saparov A, Greaves M, Welling J. Never-ending learning. Communications of the ACM, 2018, 61(5): 103−115
41. Elhammadi S, Lakshmanan L V S, Ng R, Simpson M, Huai B, Wang Z, Wang L. A high precision pipeline for financial knowledge graph construction. In: Proceedings of the 28th International Conference on Computational Linguistics. 2020, 967−977
42. Kim K, Hur Y, Kim G, Lim H. GREG: a global level relation extraction with knowledge graph embedding. Applied Sciences, 2020, 10(3): 1181
43. McCallum A, Freitag D, Pereira F C N. Maximum entropy Markov models for information extraction and segmentation. In: Proceedings of the 17th International Conference on Machine Learning. 2000, 591−598
44. Brin S. Extracting patterns and relations from the world wide web. In: Proceedings of International Workshop on the World Wide Web and Databases. 1998, 172−183
45. Hasegawa T, Sekine S, Grishman R. Discovering relations among named entities from large corpora. In: Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics. 2004, 415−422
46. Gong C, Li Z, Xia Q, Chen W, Zhang M. Hierarchical LSTM with char-subword-word tree-structure representation for Chinese named entity recognition. Science China Information Sciences, 2020, 63(10): 202102
47. Mintz M, Bills S, Snow R, Jurafsky D. Distant supervision for relation extraction without labeled data. In: Proceedings of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP. 2009, 1003−1011
48. Gormley M R, Yu M, Dredze M. Improved relation extraction with feature-rich compositional embedding models. In: Proceedings of 2015 Conference on Empirical Methods in Natural Language Processing. 2015, 1774−1784
49. Florian R, Hassan H, Ittycheriah A, Jing H, Kambhatla N, Luo X, Nicolov N, Roukos S. A statistical model for multilingual entity detection and tracking. In: Proceedings of Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics. 2004, 1−8

50. Florian R, Jing H, Kambhatla N, Zitouni I. Factorizing complex models: a case study in mention detection. In: Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics. 2006, 473−480
51. Zeng D, Liu K, Lai S, Zhou G, Zhao J. Relation classification via convolutional deep neural network. In: Proceedings of the 25th International Conference on Computational Linguistics. 2014, 2335−2344
52. Zhou P, Shi W, Tian J, Qi Z, Li B, Hao H, Xu B. Attention-based bidirectional long short-term memory networks for relation classification. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics. 2016, 207−212
53. Qiao B, Zou Z, Huang Y, Fang K, Zhu X, Chen Y. A joint model for entity and relation extraction based on BERT. Neural Computing and Applications, 2022, 34(5): 3471−3481
54. Zheng S, Wang F, Bao H, Hao Y, Zhou P, Xu B. Joint extraction of entities and relations based on a novel tagging scheme. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics. 2017, 1227−1236
55. Bekoulis G, Deleu J, Demeester T, Develder C. Joint entity recognition and relation extraction as a multi-head selection problem. Expert Systems with Applications, 2018, 114: 34−45
56. Yu X, Lam W. Jointly identifying entities and extracting relations in encyclopedia text via a graphical model approach. In: Proceedings of the 23rd International Conference on Computational Linguistics. 2010, 1399−1407
57. Li Q, Ji H. Incremental joint extraction of entity mentions and relations. In: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics. 2014, 402−412
58. Gupta P, Schütze H, Andrassy B. Table filling multi-task recurrent neural network for joint entity and relation extraction. In: Proceedings of the 26th International Conference on Computational Linguistics. 2016, 2537−2547
59. Katiyar A, Cardie C. Going out on a limb: joint extraction of entity mentions and relations without dependency trees. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics. 2017, 917−928
60. Zeng X, Zeng D, He S, Liu K, Zhao J. Extracting relational facts by an end-to-end neural model with copy mechanism. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics. 2018, 506−514
61. Miwa M, Bansal M. End-to-end relation extraction using LSTMs on sequences and tree structures. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics. 2016, 1105−1116
62. Nie Y, Tian Y, Song Y, Ao X, Wan X. Improving named entity recognition with attentive ensemble of syntactic information. In: Proceedings of Findings of the Association for Computational Linguistics. 2020, 4231−4245
63. Li Z, Ding N, Liu Z, Zheng H, Shen Y. Chinese relation extraction with multi-grained information and external linguistic knowledge. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. 2019, 4377−4386
64. Xuan Z, Bao R, Jiang S. FGN: fusion glyph network for Chinese named entity recognition. In: Proceedings of the 5th China Conference on Knowledge Graph and Semantic Computing: Knowledge Graph and Cognitive Intelligence. 2020, 28−40
65. Fu T J, Li P H, Ma W Y. GraphRel: modeling text as relational graphs for joint entity and relation extraction. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. 2019, 1409−1418
66. Shang Y M, Huang H, Mao X. OneRel: joint entity and relation extraction with one module in one step. In: Proceedings of the 36th AAAI Conference on Artificial Intelligence. 2022, 11285−11293
67. Wang Y, Yu B, Zhang Y, Liu T, Zhu H, Sun L. TPLinker: single-stage joint extraction of entities and relations through token pair linking. In: Proceedings of the 28th International Conference on Computational Linguistics. 2020, 1572−1582

Mengfan Li received the BS degree from the School of Computer Science and Engineering, Northeastern University, China in 2021. She is currently working towards the PhD degree with the School of Computer Science and Technology, Huazhong University of Science and Technology, China. Her research interests include large language models, graph neural networks, and knowledge graphs.

Xuanhua Shi is a professor in National Engineering Research Center for Big Data Technology and System/ Services Computing Technology and System Lab, Huazhong University of Science and Technology, China. He received the PhD degree in computer engineering from Huazhong University of Science and Technology, China in 2005. From 2006, he worked as an INRIA Post-Doc in PARIS team at Rennes for one year. His current research interests focus on the cloud computing and big data processing. He published over 100 peer-reviewed publications, received research support from a variety of governmental and industrial organizations, such as the National Natural Science Foundation of China, the Ministry of Science and Technology of China, the Ministry of Education of China, the European Union, Alibaba, ByteDance, and Intel. Shi is a senior member of IEEE and CCF.

Chenqi Qiao received the BS degree from the Huazhong University of Science and Technology (HUST), China in 2022. He is now working toward the MS degree in the School of Computer Science and Technology, HUST, China.

Xiao Huang is an assistant professor in the Department of Computing at The Hong Kong Polytechnic University (PolyU), China. He received BS in Engineering from Shanghai Jiao Tong University, China in 2012, MS in Electrical Engineering from Illinois Institute of Technology, USA in 2015, and PhD in Computer Engineering from Texas A&M University, USA in 2020. His research interests include large language models, graph neural networks, knowledge graphs, network anomaly detection, and recommender systems. He has received over 2,600 citations. He is a program committee member of ICLR 2022-2024, NeurIPS 2021-2023, AAAI 2021-2023, KDD 2019-2023, TheWebConf 2022-2023, ICML 2021-2023, IJCAI 2020-2023, CIKM 2019-2022, and WSDM 2021-2023. Before joining PolyU, China he worked as a research intern at Microsoft Research and Baidu USA.

Weihao Wang received the MS degree from Huazhong University of Science and Technology, China. His research interest lies in knowledge graphs.

Yao Wan received his PhD degree from the College of Computer Science, Zhejiang University, China in 2019. He is currently a lecturer at the College of Computer Science and Technology, Huazhong University of Science and Technology, China. He has been a visiting student of the University of Technology Sydney, Australia and the University of Illinois at Chicago, USA in 2016 and 2018, respectively. His research interests lie in the synergy between artificial intelligence and software engineering, especially natural language processing, programming languages, software engineering, and data mining.

Teng Zhang is an associate professor of the College of Computer Science and Technology, Huazhong University of Science and Technology, China. He received his PhD degree from the Department of Computer Science and Technology, Nanjing University, China in 2019. His research interests include machine learning, data mining, artificial intelligence, and optimization. He is a program committee member of ICML 2018-2023, NeurIPS 2018-2023, IJCAI 2017-2023, AAAI 2017-2024, KDD 2017-2023 and an associate editor of the Frontiers of Computer Science. He is the best paper award owner of 2020 IEEE BigComp.

Hai Jin received his PhD degree in computer engineering from Huazhong University of Science and Technology, China in 1994. He received German Academic Exchange Service fellowship to visit the Technical University of Chemnitz, Germany in 1996. He worked at the University of Hong Kong, China between 1998 and 2000, and as a visiting scholar at the University of Southern California, USA between 1999 and 2000. He received the Excellent Youth Award from the National Natural Science Foundation of China in 2001. He is a Cheung Kung Scholars chair professor of computer science and engineering of Huazhong University of Science and Technology, the chief scientist of ChinaGrid, the largest grid computing project in China, and the chief scientist of National 973 Basic Research Program Project of Virtualization Technology of Computing System, and Cloud Security. He has coauthored 22 books and published over 800 research papers. His research interests include computer architecture, virtualization technology, cluster computing and cloud computing, peer-to-peer computing, network storage, and network security. He is a fellow of the CCF, a fellow of the IEEE and a member of the ACM.