



Research article

Food safety in health: a model of extraction for food contaminants

Yuanyuan Cai^{1,2}, Hao Liang^{1,2}, Qingchuan Zhang^{1,2,*}, Haitao Xiong³ and Fei Tong^{1,2}

¹ National Engineering Research Centre for Agri-Product Quality Traceability, Beijing Technology and Business University, Beijing 100048, China

² China Food Flavor and Nutrition Health Innovation Center, Beijing Technology and Business University, Beijing 100048, China

³ School of International Economics and Management, Beijing Technology and Business University, Beijing 100048, China

* **Correspondence:** Email: zqc1982@126.com.

Abstract: Contaminants are the critical targets of food safety supervision and risk assessment. In existing research, food safety knowledge graphs are used to improve the efficiency of supervision since they supply the relationship between contaminants and foods. Entity relationship extraction is one of the crucial technologies of knowledge graph construction. However, this technology still faces the issue of single entity overlap. This means that a head entity in a text description may have multiple corresponding tail entities with different relationships. To address this issue, this work proposes a pipeline model with neural networks for multiple relations enhanced entity pairs extraction. The proposed model can predict the correct entity pairs in terms of specific relations by introducing the semantic interaction between relation identification and entity extraction. We conducted various experiments on our own dataset FC and on the open public available data set DuIE2.0. The results of experiments show our model reaches the state-of-the-art, and the case study indicates our model can correctly extract entity-relationship triplets to release the problem of single entity overlap.

Keywords: food safety monitoring and risk assessment; multiple relationship extraction; neural network pre-training; single entity overlap; semantic interaction

1. Introduction

Fresh food (including agricultural products, fruits, etc.) is an important part of the healthy diet

that provides essential nutrients to the body. However, with the continuous development of agriculture and industry, a large number of pollutants have been discharged into the environment, and food contaminants that are detrimental to the health of people have entered the food supply chain [1]. Some studies show that a high proportion of foodborne illnesses are associated with the ingestion of contaminated food [2]. For example, people who ingest food with excessive organophosphorus pesticide residues develop have mental abnormalities, chronic neuritis and other poisoning symptoms [3], since organophosphorus pesticide inhibits cholinesterase and blocks the transmission of neurotransmitters in the human body. Given the widespread presence of food contaminants and their serious health risks to people, preventing food contamination from injuring human health is a top public health priority around the globe [1,4].

Food safety supervision and risk assessment are effective ways to prevent contaminated food from entering the supply chain and endangering human health. However, supervision and assessment suffer from the huge number of detection samples (food) and detection items (food contaminants), which leads to inefficient supervision. To improve the efficiency of supervision, some studies utilize food knowledge graphs, which contain structured knowledge of food and contaminants. The associations between food categories and food contaminants can clearly indicate which food contaminants are vital for some foods and need to be focused supervision. Food knowledge graphs are derived from various sources, which consist of knowledge about recipes, nutrients, health and food safety [5].

Entity relation extraction is the core technology for building knowledge graphs which identify entities and relationships between entities. It includes two sub-tasks: named entity identification and relationship extraction. Named entity recognition [6] extracts entities from text and divides them into specified categories. Relation extraction [7,8] identifies some semantic relationship between entities. The approaches of entity relation extraction can be divided into the pipeline and joint approaches, according to the order of the two sub-tasks. Pipeline-based approaches [9] are widely proposed to extract entities and relationships in early studies. These studies aim to extract entity information and classify the relationships between entities by training two models, respectively. However, most of the pipeline-based approaches ignore the correlation between entity recognition and relationship prediction tasks. Thus, many joint models [10–13] are proposed in recent studies to capture the shared features of entities and relations by their joint learning. However, the joint models suffer from the global optimization problem [14]. The emergence of deep learning has accelerated the application of neural networks on entity relation extraction since the neural models can automatically learn features and thus reduce the dependence of models on manual feature selection.

In the existing studies on entity relationship extraction, the problem of overlapping triple [15,16] leads to the poor performance of approaches. As shown in Figure 1, S1 belongs to the normal class, S2 belongs to the single entity overlap (SEO) class and SEO indicates that one subject pair corresponds to multiple objects in a sentence. In order to address the issue of single entity overlap, this paper proposes a pipeline model for multiple relations enhanced entity pairs extraction. Our model consists of a multi-relationship extraction module and an entity pair extraction module, both of which are trained independently. The entity relationships learned in the multi-relationship extraction module are taken as extra features to input the entity pair extraction module. As shown in experiment results on the dataset on food domain and general domain, our model provides an effective solution for a single entity overlap problem, and it can effectively mitigate the independence between entities and relationships that exist in the pipeline models.

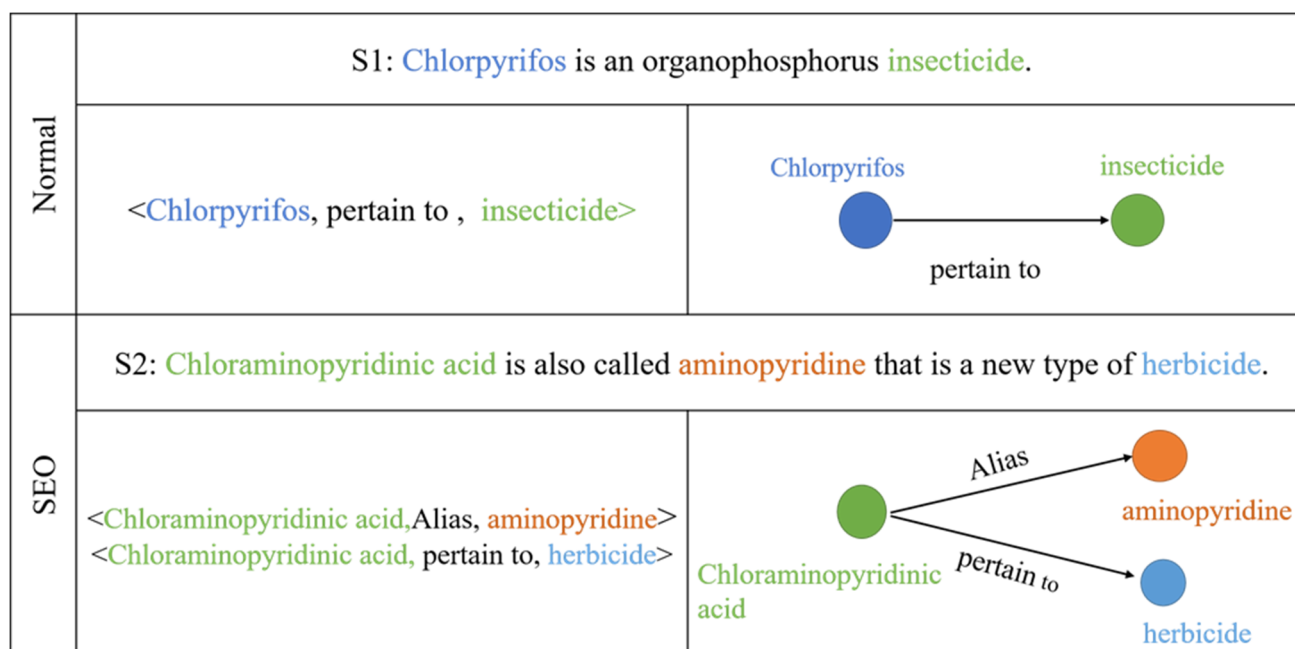


Figure 1. Examples of normal and single entity overlap (SEO).

The contributions of this paper are as follows:

- 1) We propose a multiple relations-enhanced entity extraction model which predicts entity relations and extracts corresponding entity pairs according to the relations, respectively.
- 2) The proposed model provides an effective solution for single entity overlap problem by introducing relational semantic features into the entity recognition. It can effectively mitigate the independence between entity tagging and relation classification that existed in the pipeline methods.
- 3) The experimental results for two datasets show that our model improves F1 by 4.02% to 28.74% comparing to baseline.

The rest of the article is organized as follows: Section 2 reviews the related work. Section 3 focuses on the detailed introduction of our model. Section 4 describes the experimental data set and the parameter setting of the proposed model. It also presents the experimental results and analysis. Section 5 provides the discussion of case study. Finally, Section 6 summarizes this paper.

2. Related work

Entity relationship extraction is a core task of information extraction [17,18]. The main goal of entity relationship extraction is to identify entity pairs from natural language text and determine the specific relationships between entities. Classical entity relationship extraction includes four categories: supervised, semi-supervised, weakly supervised and unsupervised. Supervised entity relation extraction is divided into feature-based and kernel-based methods.

Zhou et al. [19] used Support Vector Machine (SVM) as a classifier to study the influence of lexical, syntactic and semantic features on entity semantic relation extraction. However, supervised methods require a significant amount of time and effort for manual annotation data. Therefore, semi-supervised, weakly supervised and unsupervised methods have been proposed to solve small-scale

corpus annotation problems. Brin [20] used the Bootstrapping method for relation extraction between named entities. Craven et al. [21] first proposed the idea of weakly supervised machine learning in their study of extracting structured data from text to build a biological knowledge base. Hasegawa et al. [22] first proposed an unsupervised relationship extraction method between named entities. These methods rely on manual work for feature selection and require significant domain expertise. With the development of deep learning, neural network models have made new advances in entity relationship extraction. Zeng et al. [23] first proposed using a convolutional neural network (CNN) to extract lexical and sentence-level features for relationship extraction in 2014, and this improved the accuracy of the model.

Compared with the CNN model, the recurrent neural network (RNN) can fully consider the dependency between long-range words, and its memory function benefits the recognition of sequences. Socher et al. [24] first used the RNN approach for entity relationship extraction. Due to the gradient disappearance and gradient explosion problems, traditional RNNs rarely deal with long-term dependence problems in practice. Long Short-Term Memory (LSTM) is a particular type of RNN that solves these problems through three gating operations and cell states. LSTM has achieved great success in nature language processing [25,26]. Xu et al. [27] proposed an LSTM-based approach for relation extraction that incorporates several features and uses a maximum pooling layer and a softmax layer for relation classification. In addition, Bidirectional Encoder Representation from Transformers (BERT), as a pre-trained model, is increasingly used in relation extraction. Shi et al. [28] proposed a simple BERT model for relationship extraction.

The methods of entity relation extraction can also be divided into the pipeline and joint methods, according to the difference in the order of the two sub-tasks of entity recognition and relationship extraction. The pipeline-based methods of entity relation extraction use entity identification and relationship extraction as two subtasks, using one model to identify entities and the other to classify relationships. Xu et al. [29] improved Zeng's work by proposing a CNN based on a dependency analysis tree to extract entity relationships. The model passes the input text through the dependency analysis tree and proposes a negative sampling strategy to solve the problem of irrelevant information introduced by the dependency analysis tree when entities are distant from each other.

Santos et al. [30] propose the CR-CNN model, which uses a new pairwise ranking loss function in the output layer to replace the softmax loss function. Lin et al. [31] proposed a sentence-level attention-based CNN for relation extraction; it introduced an attention mechanism at the sentence level to reduce the influence of noisy sentences, effectively improving cross-linguistic consistency and complementarity. Zhang et al. [32] used a Bi-directional Long Short-Term Memory (BiLSTM) model, combining information before the current word and after the word, for relationship extraction. Chen et al. [14] presented a simple pipeline approach for entity and relation extraction; their model builds on span-level representations and fuses entity information (including boundaries and types) in the input layer of the relational model.

Joint models aim to extract entities and relations simultaneously, in two different ways, based on multi-task learning and structured prediction. In the multi-task learning-based entity relation extraction method, the two sub-tasks of entity identification and relationship extraction are learned jointly by sharing the encoding layer of the joint model. Miwa et al. [33] used an LSTM-based model to extract entities and relations, which employs a neural network to reduce manual work. Li et al. [11] presented an incremental joint framework to extract entity mentions and relations using a structured perceptron with efficient beam search. Zheng et al. [34] fed sentences into an embedding layer and a Bi-LSTM

layer, using an LSTM for named entity recognition and a CNN for relation extraction. Xue et al. [35] presented a focused attention model for the joint entity and relation extraction task that integrates the BERT language model as a shared parameter layer. Bekoulis et al. [36] proposed a joint neural model that models the entity recognition task using a conditional random field layer and the relationship extraction task as a multi-headed selection problem (i.e., multiple relationships may be identified for each entity).

The structured prediction way is direct modeling of entity-relationship triples. Zheng et al. [37] proposed an entity relationship extraction method based on a new annotation strategy, which turned the original joint learning model involving two subtasks of named entity recognition and relationship classification into a sequence labeling problem. Katiyar et al. [38] first used the attention mechanism with BiLSTM for joint entity extraction and classification of relations. The model improves the deficiencies of Miwa's work [33], which relies on features such as lexical labels and dependent trees in the relationship classification sub-task. Li et al. [39] proposed a paradigm to transform entity and relation extraction into a multi-turn question-answering (QA) task.

However, most sentences have the problem of entity overlap. To solve this problem, Wei et al. [15] proposed a cascading pointer labeling approach. Zeng et al. [16] proposed an end-to-end neural model based on sequence-to-sequence learning with copy mechanism to extract the entities and relations. Dai et al. [40] solved the entity overlap problem by designing a particular marking scheme and introducing a position-attention mechanism. Fu et al. [41] presented a model based on graph convolutional networks (GCNs) to extract entities and relations, which can also effectively overlap problem. Eberts et al. [42] introduced a span-based attention model SpERT for entity and relation extraction.

3. Methodology

In this section, we introduce the structure of modeling multiple relations enhanced entity pairs extraction for the food contaminants domain. As shown in Figure 2, our model includes the multi-relation extraction module and the entity pair extraction module.

The multi-relation extraction module mainly includes ALBERT [43] pre-training network and TextCNN [44,45] network. The module trains the input sentences to obtain the text feature vector and uses sigmoid activation function to output the predicted relationship. The detail of this module is described in Section 3.1.

The entity pair extraction module mainly includes the RoBERTa [46], BiLSTM and CRF layers. In this module, we take the output of the multi-relation extraction module as an input of this module to strengthen the correlation between relationships and entities. The module uses Roberta to obtain the vector representation of the sentence. After splicing with the relational vector, it is trained as the input of BiLSTM and obtains the hidden layer vector. Finally, the BIO labels are finally output through the CRF layer to indicate the entity pairs. As shown on the right side of Figure 2, the sentence contains two different relationships, and the entity pairs corresponding to each relationship are printed out. The detail of this module is described in Section 3.2.

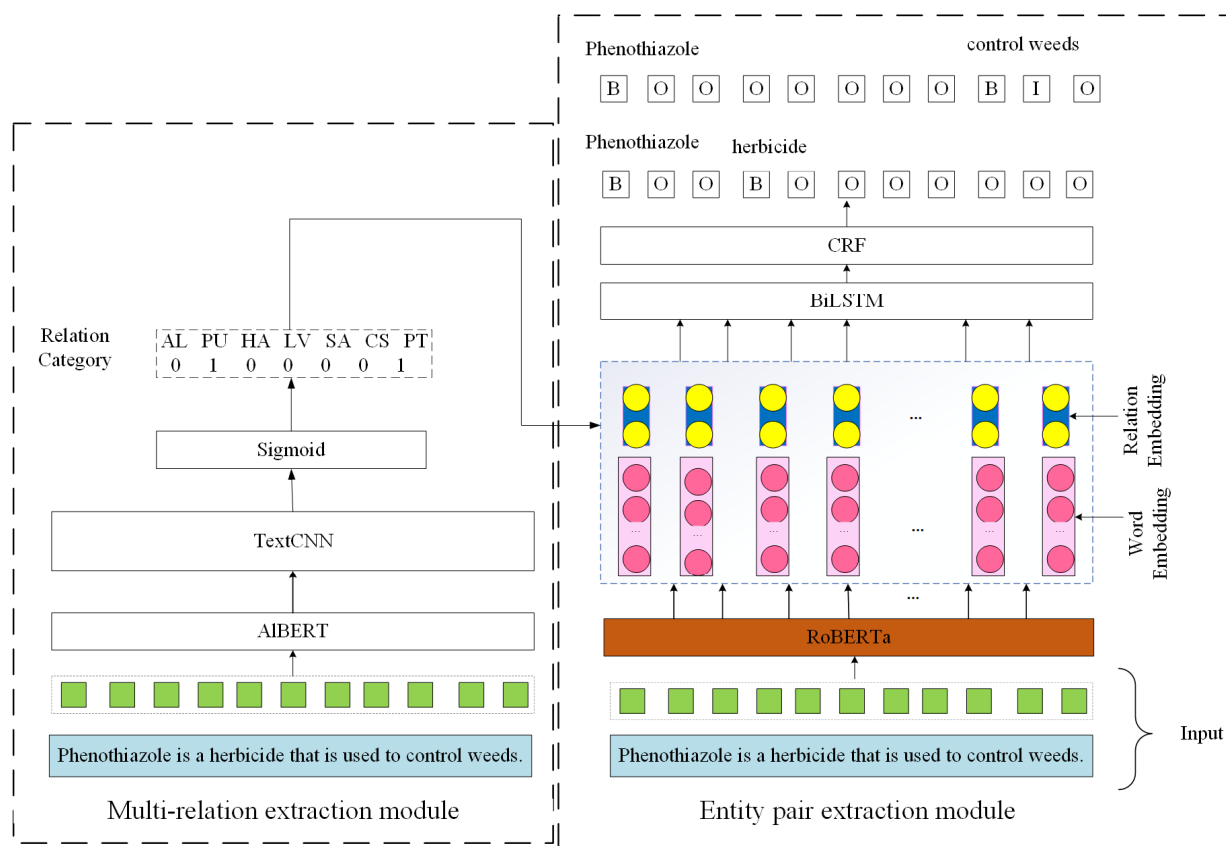


Figure 2. The framework of our model.

ALBERT and RoBERTa both use Transformer-based encoders, where the main module is self-attention. Self-attention was proposed to get rid of the drawback that recurrent neural networks cannot compute in parallel, and it is designed in such a way that the current word can be viewed by other words. The computational complexity of self-attention is $O(n^2 \cdot d)$, where n denotes the sequence length and d denotes the vector dimension.

3.1. Multi-relation extraction module

The multi-relation extraction module is shown in Figure 3. We applied the ALBERT and TextCNN models to identify sentence relationship types. The input sentence X passes through the ALBERT layer to obtain the feature vector, as shown in Eq (1):

$$T = [t_1, t_2, t_3, \dots, t_i, \dots, t_n] \quad (1)$$

where t_i denotes the vector representation of the i -th word of the input sentence.

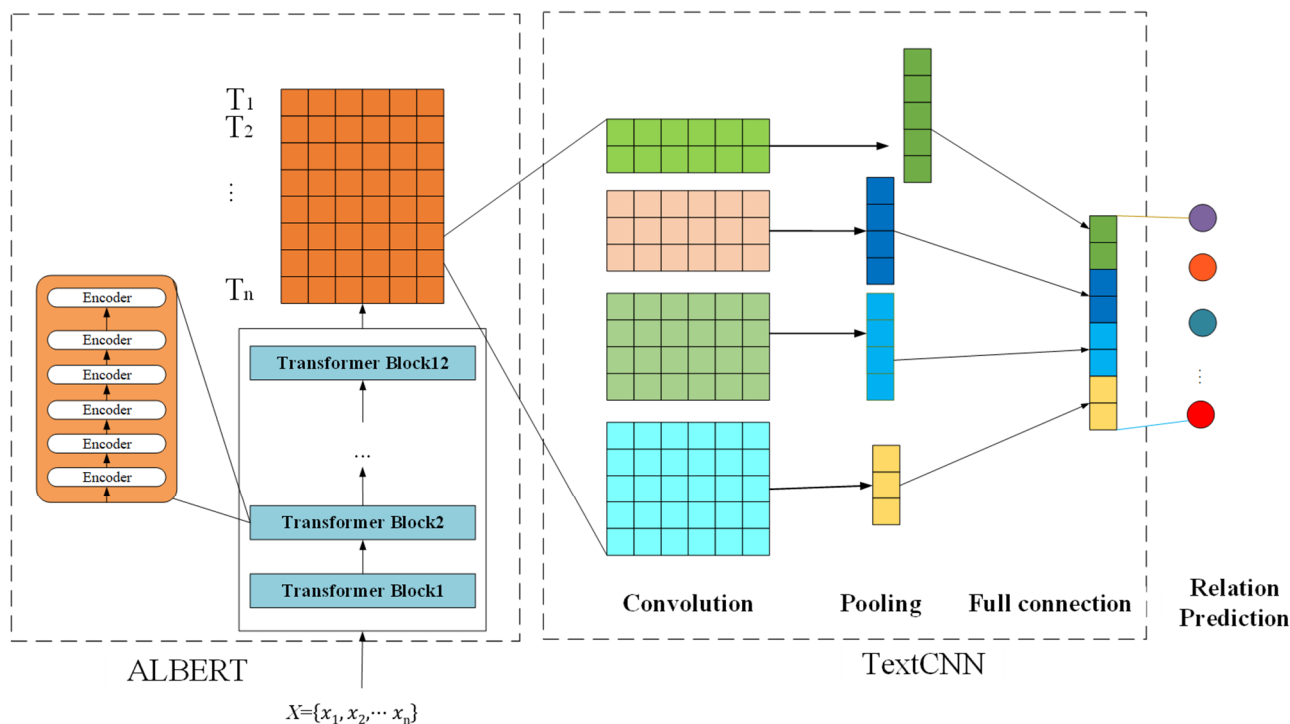


Figure 3. The framework of the multi-relation extraction module.

ALBERT is a lightweight network model based on the BERT [47] pre-trained language model. It uses a transformer encoder [48,49] with GELU nonlinearities [50]. ALBERT improves BERT in decomposition embedding parameterization, cross-layer parameter sharing and inter-sentence coherence loss. Compared to BERT, where the word embedding dimension E and the implicit dimension H is equal, ALBERT uses a factorization to reduce the number of parameters in the embedding mapping module, i.e., the one-hot vector of words is mapped to a low-dimensional (size E) space and then mapped back to a high-dimensional (size H) space, which reduces the number of parameters from $O(V \times H)$ to $O(V \times E + E \times H)$. When $E \ll H$, the reduction of the parameter is pronounced.

The TextCNN network receives the feature vector output from ALBERT as input. The convolution layer extracts the feature information of different granularities in the semantic feature vector. This method can be implemented by setting different convolution kernel sizes. Different convolution kernel sizes get different feature set sizes. In this paper, the sizes of the convolution kernel are set to 2, 3, 4 and 5. The pooling function is used for each feature set, and max pooling is used to extract the maximum value in the feature set in the feature collection. Finally, all the output feature values are stitched to get the text feature vector representation.

The full connection layer employs the sigmoid function to output the sentence relations, as shown in Eq (2):

$$\text{sigmoid} = \frac{1}{1+e^{-x}} \quad (2)$$

3.2. Entity pair extraction module

The entity pair extraction module is shown in the right half of Figure 2. RoBERTa is a robustly optimized BERT pretraining approach. The BERT model is composed of a multi-layer bidirectional transformer from which the encoder can learn practical information in the context well. RoBERTa improves upon BERT by adopting a dynamic mask mechanism, eliminating the next sentence prediction task and using byte-level-based byte-Pair Encoding (BPE) encoding for text encoding. The advantage of using bytes-level BPE encoding is that the method uses bytes as the base word unit, allowing arbitrary text to be encoded without introducing unknown characters. We applied a pre-trained RoBERTa model to encode the input sentence S . Formally, a sentence of length n can be represented as $S = \{w_1, w_2, w_3, \dots, w_n\}$. We input these tokens into the same RoBERTa encoder layer as the trained one to obtain a vector representation H of the input sentence S , as follows in Eq (3):

$$H = [h_1, h_2, h_3, \dots, h_n] = \text{RoBERTa}(S) \quad (3)$$

where S denotes the input sentence, H represents the text vector and h_i means the vector representation of the i -th word after RoBERTa encoding.

The Bidirectional Long Short-Term Memory (BiLSTM) [40] model takes word embedding and relation embedding as input, in which word embedding is learned by BERT and relation embedding is learned by the multi-relation extraction module. We take a relation obtained in 3.1 to construct the relation vector V , as shown in Eq (4):

$$V = [v_1, v_2, v_3, \dots, v_i, \dots, v_t] \quad (4)$$

We built a t -dimension relation vector which each dimension represents a kind of relation. Where v_i represents the i -th dimension and each dimension can be 0 or 1. 1 means that the i -th relation exists, and 0 means that the relation does not exist.

Finally, H and V are concatenated into the mixture vector M , as shown in Eq (5):

$$M = [H:V] \quad (5)$$

where: indicates the splicing of vectors.

BiLSTM is the abbreviation of Bidirectional Long Short-Term Memory, a combination of forward LSTM and backward LSTM. Usually, the LSTM network encodes sentences from front to back and only masters the context information from front to back and does not master the context information from back to front, so the forward LSTM network and the backward LSTM network are formed into a BiLSTM network to learn two-way context information [51]. The output of the BiLSTM layer is expressed as P_{mn} where n represents the number of words in the sentence, and m represents the label type.

The Conditional Random Field (CRF) [52] is a discriminative model that focuses on solving the problem of serialized annotation. In recent years, it has been widely used in word segmentation, part-of-speech tagging, named entity recognition and other sequence tagging work, and it has achieved good results [53]. CRF combines the advantages of discriminative models while considering the probability of transfer between contextual tokens and the characteristics of global parameter optimization and decoding in serialized form. For the input sentence sequence $x = \{x_1, x_2, \dots, x_n\}$ and its predicted sequence $y = \{y_1, y_2, \dots, y_n\}$, the basic algorithm of CRF is defined as shown in Eq (6),

$$S(x, y) = \sum_{i=1}^n A_{y_{i-1}, y_i} + \sum_{i=0}^n P_{i, y_i} \quad (6)$$

where P_{i, y_i} represents the probability that the i -th matches the y_i -th label. A is the transfer matrix, A_{y_{i-1}, y_i} denotes the score from label y_{i-1} to label y_i . After normalization all possible outputs, the probability distribution about the output sequence y is obtained, as shown in Eq (7),

$$P(y|x) = \frac{e^{S(x, y)}}{\sum_{\tilde{y} \in Y_x} e^{S(x, \tilde{y})}} \quad (7)$$

In the training process, we have taken the likelihood function value of the maximally correct label sequence, as shown in Eq (8),

$$\log(p(y^* | S)) = S(x, y^*) - \log(\sum_{\tilde{y} \in Y_x} e^{S(x, \tilde{y})}) \quad (8)$$

where Y_x represents the result of all possible output tag sequences of sentence X . Using the likelihood function for calculation, we can obtain a valid sequence of labels. Then, the output sequence with the highest overall probability is the predicted result as shown in Eq (9),

$$y^* = \arg \max S(x, \tilde{y}) \quad (9)$$

We choose the Viterbi algorithm [54] to perform the calculation. All the outputs of the BiLSTM layer will be used as inputs to the CRF layer. The CRF layer can add some constraints to the final predicted label to ensure that the predicted label is valid. During training, these constraints can be automatically learned through the CRF layer. For example, according to the BIO labeling rules, the label of the first word in each sentence must be “B-” or “O-”; it cannot be “I-” because “B” indicates the first word of an entity, “I” indicates the second word and the word after the entity, and “O” indicates a word that does not belong to a specific entity. In addition, a combination similar to “B-I-OBJ” must be wrong because the adjacent “B-” and “I-” must represent the same type of entity. Based on these rules, it is possible to obtain a higher accuracy rate in prediction.

4. Experiments and results analyses

In this section, we introduce our experimental environment and parameter settings. Then, we compare the experimental results of different models for entity relation extraction.

4.1. Data set and data processing

We applied the food contaminant data constructed separately in the field of food safety as the experimental data set FC. A self-built corpus collects data from professional and authoritative websites on food contaminants (such as State Administration Market Regulation, Baidu Encyclopedia and Foodmate Net) by Crawler. In addition, the open data set DuIE2.0 has forty-eight relations. It was applied for comparison experiments to ensure the fairness of the experimental results. The FC dataset consists of seven entity relations, which are pre-defined. Table 1 presents the seven relation types along with their names and abbreviations.

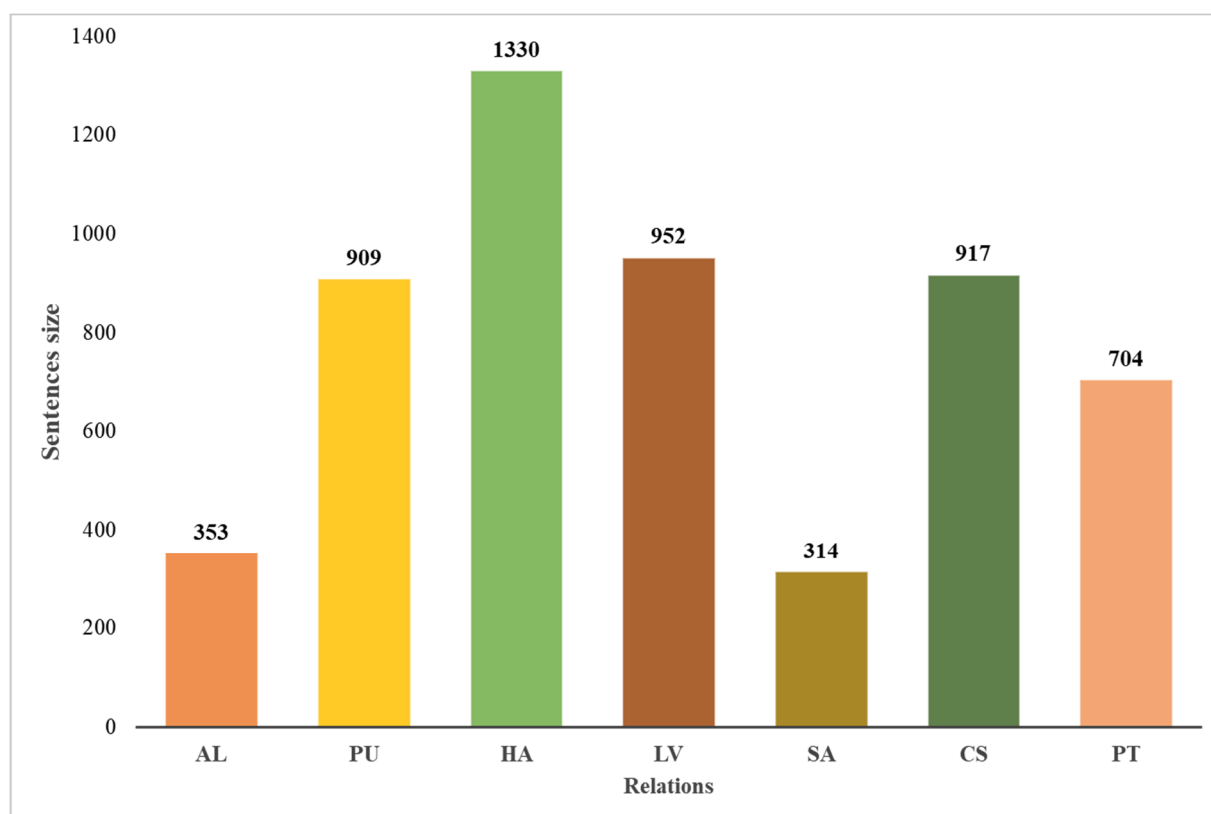
Table 1. Definition of relationships for the FC dataset.

ID	Relation name	Abbreviations
1	Alias	AL
2	Purpose	PU
3	Harm	HA
4	Limit value	LV
5	Scope of application	SA
6	Corresponding standards	CS
7	Pertain to	PT

The dataset is divided into three parts: the training set, the validation set and the test set. The size settings of datasets are shown in Table 2. Additionally, the sentence size of each relationship in the experimental FC is shown in Figure 4.

Table 2. Data set size statistics.

Data Set	Training	Validation	Test	Label
FC	1400	200	400	7
DuIE2.0	170,000	20,000	20,000	48

**Figure 4.** Details of the FC experimental data set.

The data Pre-processing is shown in Figure 5. We labeled seven types of relationships. We

adopted seven bits to indicate whether there is a correlation between two entities, where 1 indicates correlation and 0 indicates no correlation. The corpus has been marked according to the ternary mark {B, I, O}. B represents the first word of the entity, I represent the second word and the following words of the entity and O represent the word that does not belong to a specific entity. At the same time, we also annotate the entity type; SUB indicates head entity and OBJ indicates tail entity.

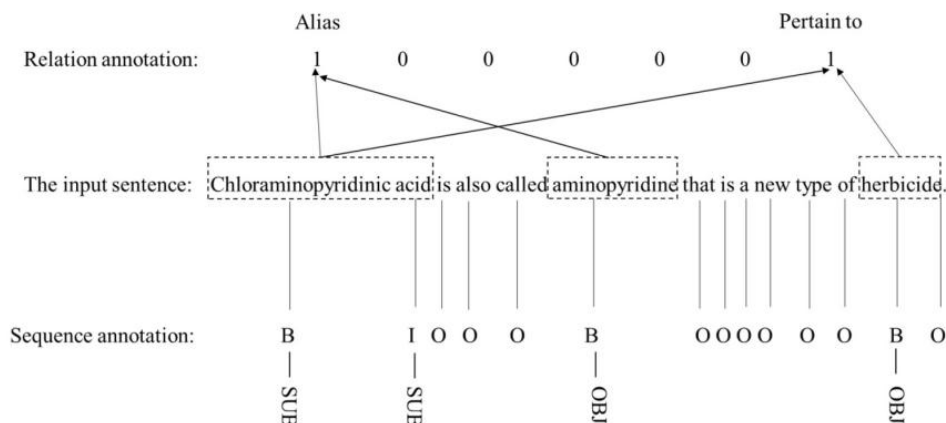


Figure 5. Manual annotation.

4.2. Evaluation criterion

We applied three common indicators, precision (P), recall (R) and F1, to evaluate our model and baseline models. The precision calculation formula is as follows in Eq (10),

$$P = \frac{TP_i}{TP_i + FP_i} \quad (10)$$

where the precision is referred to as P. TP_i represents the number of positive classes predicted by the model correctly and FP_i represents the number of positive classes predicted by the model from negative classes. The recall calculation formula is as follows in Eq (11),

$$R = \frac{TP_i}{TP_i + FN_i} \quad (11)$$

where the recall is referred to as R. TP_i is the same as the above-mentioned formula and FN_i represents the number of negative classes predicted by the model from positive classes. The F1 calculation formula is as follows in Eq (12),

$$F1 = \frac{2 * P * R}{P + R} \quad (12)$$

where F1 represents the harmonic mean of precision and recall.

4.3. Implementation parameters

The main experimental parameters are set as Table 3, RoBERTa model has 12 hidden layers and the dimension of each hidden layer is 768; the hidden size of the BiLSTM model is 128. We used the open-source deep learning framework TensorFlow to build our neural network model.

Table 3. Experimental parameter setting.

Model	Parameters
RoBERTa	12-layer
	768-dimensions
	learning rate 1e-3
	pad size 128
	Tanh function
BiLSTM	2-layer
	128-dimensions
	learning rate 1e-3
	pad size 128
	ReLU function
	Tanh function
	2-layer

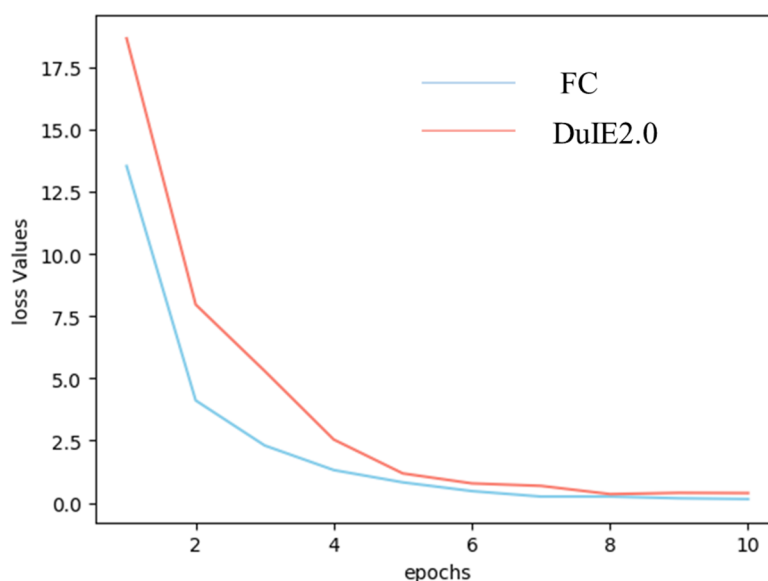


Figure 6. The loss of different epochs.

Experimental hyperparameters settings are slightly different for different data sets. In order to choose optimal epochs, we conducted experiments to show the trend of loss value under different epochs. Figure 6 shows the relationship between epochs and loss value. As the number of epochs increases, the loss value decreases rapidly to a more stable state. According to Figure 6, we set the max times of epoch as 10. We wielded Adam [53] to optimize parameters. The model is trained until the

loss value reaches a threshold or the epoch times reach the maximum.

4.4. Analysis of experimental results

In the multi-relation extraction module, we employed the seven annotated relations indicated in Table 1 to conduct comparative experiments between FC and DuIE2.0. The experimental results of relation identification for each model, including accuracy, recall and F1 value, are shown in Table 4.

Table 4. Experimental results of multi-relation extraction.

Model	FC			DuIE2.0		
	P%	R%	F1%	P%	R%	F1%
TextCNN [44,45]	68.77	69.22	68.99	66.23	67.51	66.87
ALBERT [43]	90.42	93.25	91.81	90.23	92.76	91.47
ALBERT-Denses [45]	89.7	89.01	89.35	88.82	89.53	89.17
ALBERT-TextCNN	94.14	94.88	94.51	93.24	94.91	94.07

The experimental results in Table 4 show that the ALBERT-TextCNN model achieved the best performance on both FC and DuIE2.0 datasets. Compared to the TextCNN model, ALBERT has more powerful linguistic representation and feature extraction capabilities and can better access the semantic information of the context. Furthermore, compared with ALBERT, adding TextCNN after ALBERT can better capture local feature information, thus further improving the accuracy of experimental results. By comparing the experimental results of the ALBERT and ALBERT-Denses models, we also found that the multi-relational classification problem was transformed into multiple binary classification problems when the output changed the fully connected layer. In this case, the experiments' accuracy decreased instead, indicating that it is more effective to use the cross-entropy loss mechanism on the multi-classification problem.

In the entity pair extraction experiment, we compared the results of different entity extraction models, as shown in Table 5.

Table 5. Experimental results of entity pair extraction.

Model	FC			DuIE2.0		
	P%	R%	F1%	P%	R%	F1%
IDCNN [55]-CRF	57.44	72.59	64.13	53.83	46.76	50.05
BiLSTM [56]-CRF	61.40	70.54	65.65	83.16	51.09	63.29
BERT	87.10	90.67	88.85	83.23	87.49	85.31
RoBERTa	87.97	91.58	89.74	84.81	87.99	86.37
Our model	93.87	93.71	93.80	92.19	92.94	92.56

The experimental results show that our model achieved better results on both FC and DuIE2.0, significantly improving precision and recall, while F1 also provided the best performance. Compared with the baseline model, the F1 value of our model increased by 4.06% and 6.19%, respectively. We used the extracted relationships as input for the entity pair extraction module, which enhanced the entity features and improved our model's overall performance. BiLSTM can better capture the bi-

directional semantic dependencies in the sentence and obtain the maximum output score for the word itself, thus improving the performance. The output sequence of BiLSTM is based on the maximum value of the current word score. However, it was found in the experiments that some words were easily subdivided or did not consider the logical constraints of adjacent tags. In these cases, the results of BiLSTM identification are not ideal. The CRF layer can add some constraints to the final prediction tag to ensure that they are valid, and these constraints can be obtained from the training data set during the training process.

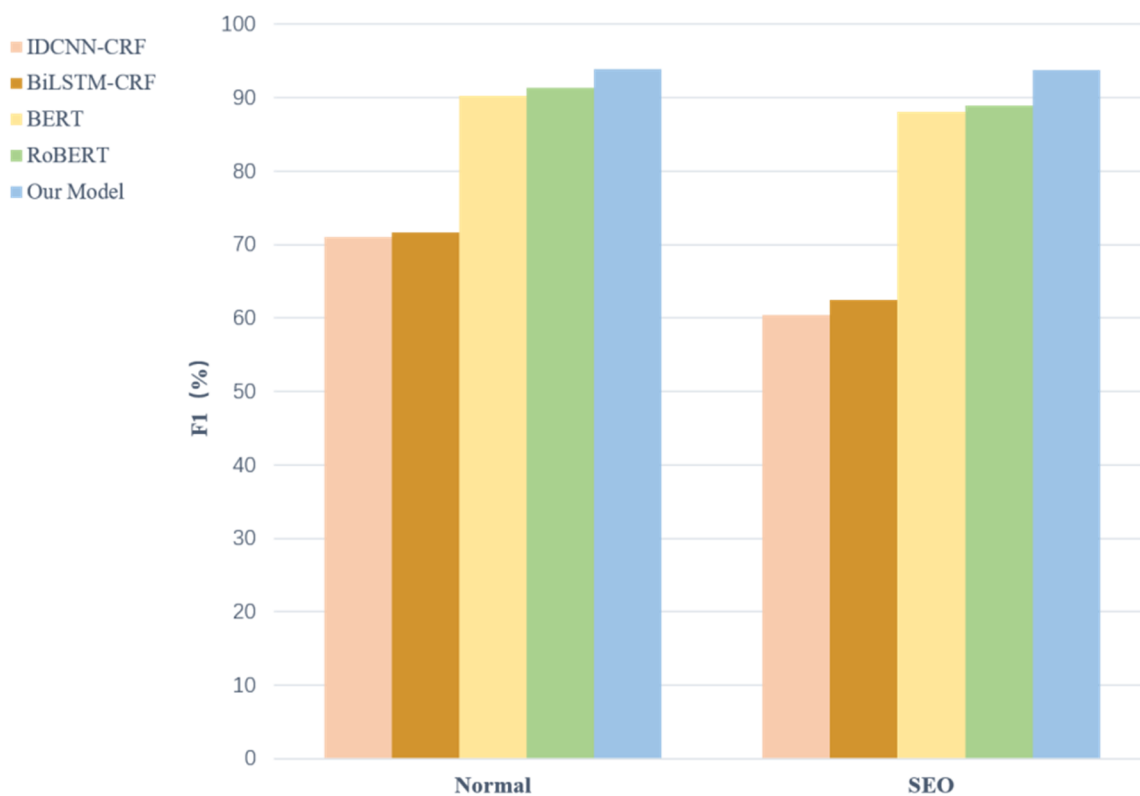


Figure 7. F1 values of compared models.

To further validate the performance of our proposed model in extracting overlapping triples, we conducted further experiments. The dataset FC was divided into Normal and SEO, where SEO contains the triplet type shown in Figure 1. As shown in Figure 7, the F1 values of all models are higher than the SEO when extracting sentences of Normal. This phenomenon indicates that as the complexity of sentences increases, the difficulty of extracting triples in sentences also rises, and the performance of the model decreases. In contrast, our model performs best when extracting different types of triples.

4.5. Ablation experiment

In order to further illustrate that the proposed model is more effective in entity extraction, we conducted an ablation experiment on our proposed model. The experimental results are shown in Table 6.

Table 6. Experimental results for ablation of entity pair extraction.

Model	FC			DuIE2.0		
	P%	R%	F1%	P%	R%	F1%
Our model	93.87	93.71	93.80	92.19	92.94	92.56
(-CRF)	90.26	90.11	90.19	89.09	89.75	89.42
(-BiLSTM)	91.14	90.98	91.06	89.70	90.45	90.07
(-BiLSTM-CRF)	87.97	91.58	89.74	84.81	87.99	86.37

It is obvious that if the CRF network is removed from our model, F1 decreases by 3.61% and 3.14% on the FC and DuIE2.0 datasets, respectively. If the BiLSTM network is removed from our model, F1 decreases by 2.74% and 2.49% on the two datasets. If the BiLSTM and CRF networks are removed from our model, F1 decreases the most on both datasets, indicating that both BiLSTM and CRF play an active role in the entity relationship extraction task. BiLSTM can combine information from the input sequence forward and backward, which helps to improve the efficiency of entity recognition. In addition, we observed that the F1 values were significantly lower after removing CRF from the model than after removing BiLSTM, indicating that CRF played a more significant role in entity recognition. Since entities require directionality for the extraction task, the role of the CRF layer is to provide a label constraint relationship for the model to ensure that labels are valid.

5. Case study and discussion

This section shows the case study of our model and its variant, as shown in Table 7. The first row in Table 7 shows an example of entity relation triples in which two entities have a single relationship. The second row shows an example with entity overlap, which in ground truth has three triples corresponding to two relationships.

For the first example, all models can correctly extract entity pair and their relationships (the blue fonts indicate that the predicted result is consistent with the ground truth). For the second example, our model could extract three entity relationship triples corresponding to two relationships. However, our model without relation enhancement could not correctly identify entity relationship triples (the red fonts indicate that the predicted result is inconsistent with the ground truth), since they only could extract a single relation.

Our model can effectively realize multi-relation identification and entity pair extraction and can solve the problem of missing information interaction in the traditional pipeline method. However, this paper does not consider the case of entity pair overlap (EPO). In the future, we will study how to solve EPO problem in entity extraction and transfer the proposed method to other application fields [57], such as biomedical and smart city, and decision support systems [58]. In addition, we will consider extending our approach from sentence-level entity relationship extraction to document-level entity relationship extraction.

Table 7. Case study of our model.

Sentence	Our Model	Our Model (without relation enhancement)	Ground Truth
Chlorpyrifos is an organophosphorus insecticide.	(Chlorpyrifos, Pertain to, insecticide)	(Chlorpyrifos, Pertain to, insecticide)	(Chlorpyrifos, Pertain to, insecticide)
Mandarin can put color on bread and biscuits.	(Mandarin, Purpose, put color on)	(Mandarin, Purpose, bread)	(Mandarin, Purpose, put color on)
	(Mandarin, Scope of application, bread)	(Mandarin, Purpose, biscuits)	(Mandarin, Scope of application, bread)
	(Mandarin, Scope of application, biscuits)	(Mandarin, Scope of application, biscuits)	(Mandarin, Scope of application, biscuits)

6. Conclusions

This paper proposes a multiple relations-enhanced entity pairs extraction model that divides the overlapping entity relation extraction task into pipeline modules that contain multi-relation extraction and entity pair extraction. In the multi-relation extraction module, multiple entity relations in sentences are correctly extracted by building the ALBERT model and TextCNN model. In the entity pair extraction module, the extracted entity relations are applied as the input of RoBERTa-BiLSTM-CRF model to accurately extract entity pairs. Experimental results on two datasets show that our model outperforms baseline models. The entity triples extracted by the model help to construct the knowledge map in the field of food safety.

Acknowledgments

This work was supported by the National Key R&D Program of China (Grant No. 2019YFC1606401), the Project of Cultivation for Young Top-notch Talents of Beijing Municipal Institutions (Grant No. BPHR202203061), the R&D Program of Beijing Municipal Commission of Education (Grant No. KM202010011011), the Humanity and Social Science Youth Foundation of Ministry of Education of China (Grant Nos. 20YJCZH229 and 21YJCZH186) and the National Natural Science Foundation of China (Grant No. 72171004).

Conflict of interest

The authors declare there is no conflict of interest.

References

1. W. Guo, B. Pan, S. Sakkiyah, G. Yavas, W. Ge, W. Zou, et al., Persistent organic pollutants in food: contamination sources, health effects and detection methods, *Int. J. Environ. Res. Public Health*, **16** (2019), 4361. <https://doi.org/10.3390/ijerph16224361>
2. F. Yeni, S. Yavaş, H. Alpas, Y. Soyer, Most common foodborne pathogens and mycotoxins on fresh produce: a review of recent outbreaks, *Crit. Rev. Food Sci. Nutr.*, **56** (2016), 1532–1544. <https://doi.org/10.1080/10408398.2013.777021>
3. C. A. Damalas, I. G. Eleftherohorinos, Pesticide exposure, safety issues, and risk assessment indicators, *Int. J. Environ. Res. Public Health*, **8** (2011), 1402–1419. <https://doi.org/10.3390/ijerph8051402>
4. P. Bertail, S. Cléménçon, J. Tressou, A storage model with random release rate for modeling exposure to food contaminants, *Math. Biosci. Eng.*, **5** (2008), 35–60. <https://doi.org/10.3934/mbe.2008.5.35>
5. W. Min, C. Liu, L. Xu, S. Jiang, Applications of knowledge graphs for food science and industry, *Patterns*, **3** (2022), 100484. <https://doi.org/10.1016/j.patter.2022.100484>
6. C. Li, K. Ma, Entity recognition of Chinese medical text based on multi-head self-attention combined with BILSTM-CRF, *Math. Biosci. Eng.*, **19** (2022), 2206–2218. <https://doi.org/10.3934/mbe.2022103>
7. H. Yu, H. Li, D. Mao, Q. Cai, A domain knowledge graph construction method based on Wikipedia, *J. Inf. Sci.*, **47** (2021), 783–793. <https://doi.org/10.1177/0165551520932510>
8. H. Yu, H. Li, D. Mao, Q. Cai, A relationship extraction method for domain knowledge graph construction, *World Wide Web*, **23** (2020), 735–753. <https://doi.org/10.1007/s11280-019-00765-y>
9. K. Hashimoto, M. Miwa, Y. Tsuruoka, T. Chikayama, Simple customization of recursive neural networks for semantic relation classification, in *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (ACL)*, (2013), 1372–1376.
10. Q. Li, H. Ji, Incremental joint extraction of entity mentions and relations, in *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL)*, (2014), 402–412. <https://doi.org/10.3115/v1/P14-1038>
11. X. Yu, W. Lam, Jointly identifying entities and extracting relations in encyclopedia text via a graphical model approach, in *International Conference on Computational Linguistics*, (2010), 1399–1407. Available from: <https://aclanthology.org/C10-2160>.
12. H. Chang, H. Zan, T. Guan, K. Zhang, Z. Sui, Application of cascade binary pointer tagging in joint entity and relation extraction of Chinese medical text, *Math. Biosci. Eng.*, **19** (2022), 10656–10672. <https://doi.org/10.3934/mbe.2022498>
13. Z. Liang, Z. Zhang, H. Chen, Z. Zhang, Disease prediction based on multi-type data fusion from Chinese electronic health record, *Math. Biosci. Eng.*, **19** (2022), 13732–13746. <https://doi.org/10.3934/mbe.2022640>
14. Z. Zhong, D. Chen, A frustratingly easy approach for entity and relation extraction, in *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics Human Language Technologies (NAACL)*, (2021), 50–61. <https://doi.org/10.18653/v1/2021.naacl-main.5>

15. X. Zeng, D. Zeng, S. He, K. Liu, J. Zhao, 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 (ACL)*, (2018), 506–514. <https://doi.org/10.18653/v1/P18-1047>
16. Z. Wei, J. Su, Y. Wang, Y. Tian, Y. Chang, A novel cascade binary tagging framework for relational triple extraction, *arXiv preprint*, (2010), arXiv:1909.03227. <https://doi.org/10.48550/arXiv.1909.03227>
17. Y. Zhang, X. Li, Y. Yang, T. Wang, Disease- and drug-related knowledge extraction for health management from online health communities based on BERT-BiGRU-ATT, *Int. J. Environ. Res. Public Health*, **19** (2022), 16590. <https://doi.org/10.3390/ijerph192416590>.
18. Q. Pan, C. Huang, D. Chen, A method based on multi-standard active learning to recognize entities in electronic medical record, *Math. Biosci. Eng.*, **18** (2021), 1000–1021. <https://doi.org/10.3934/mbe.2021054>
19. G. Zhou, J. Su, J. Zhang, M. Zhang, Exploring various knowledge in relation extraction, in *Proceedings of the Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics (ACL)*, (2005), 427–434. <https://doi.org/10.3115/1219840.1219893>
20. S. Brin, Extracting patterns and relations from the World Wide Web, in *the World Wide Web and Databases*, Springer, (1999), 172–183. https://doi.org/10.1007/10704656_11
21. M. Craven, J. Kumlien, Constructing biological knowledge bases by extracting information from text Sources, *Proc. Int. Conf. Intell. Syst. Mol. Biol.*, **1999** (1999), 77–86.
22. T. Hasegawa, S. Sekine, R. Grishman, Discovering relations among named entities from large corpora, in *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics*, (2004), 415–422. <https://doi.org/10.3115/1218955.1219008>
23. D. Zeng, K. Liu, S. Lai, G. Zhou, J. Zhao, Relation classification via convolutional deep neural network, in *Proceedings of COLING 2014 the 25th International Conference on Computational Linguistics: Technical Papers (COLING)*, (2014), 2335–2344.
24. R. Socher, B. Huval, C. D. Manning, A. Y. Ng, Semantic compositionality through recursive matrix-vector spaces, in *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP)*, (2012), 1201–1211.
25. J. R. Barr, P. Shaw, F. N. Abu-Khzam, S. Yu, H. Yin, T. Thatcher, Combinatorial code classification & vulnerability rating, in *Second International Conference on Transdisciplinary AI (TransAI)*, (2020), 80–83, <https://doi:10.1109/TransAI49837.2020.00017>
26. K. T. Chui, B. B. Gupta, P. Vasant, A genetic algorithm optimized RNN-LSTM model for remaining useful life prediction of turbofan engine, *Electronics*, **10** (2021), 285. <https://doi.org/10.3390/electronics10030285>
27. Y. Xu, L. Mou, G. Li, Y. Chen, H. Peng, Z. Jin, Classifying relations via long short term memory networks along shortest dependency paths, in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, (2015), 785–1794. <https://doi.org/10.18653/v1/D15-1206>
28. P. Shi, J. Lin, Simple BERT models for relation extraction and semantic role labeling, *arXiv preprint*, (2019), arXiv:1904.05255. <https://doi.org/10.48550/arXiv.1904.05255>
29. K. Xu, Y. Feng, S. Huang, D. Zhao, Semantic relation classification via convolutional neural networks with simple negative sampling, *arXiv preprint*, (2015), arXiv:1506.07650. <https://doi.org/10.48550/arXiv.1506.07650>

30. C. Santos, B. Xiang, B. Zhou, Classifying relations by ranking with convolutional neural networks, in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, (2015), 626–634. <https://doi.org/10.48550/arXiv.1504.06580>
31. Y. Lin, S. Shen, Z. Liu, H. Luan, M. Sun, Neural relation extraction with selective attention over instances, in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL)*, **1** (2016), 2124–2133. <https://doi.org/10.18653/v1/P16-1200>
32. S. Zhang, D. Zheng, X. Hu, M. Yang, Bidirectional long short-term memory networks for relation classification, in *Proceedings of the 29th Pacific Asia Conference on Language Information and Computation (PACLIC)*, (2015), 73–78.
33. M. Miwa, M. Bansal, End-to-end relation extraction using LSTMs on sequences and tree structures, in *Proceedings of the Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL)*, **1** (2016), 1105–1116. <https://doi.org/10.18653/v1/P16-1105>
34. S. Zheng, Y. Hao, D. Lu, H. Bao, J. Xu, H. Hao, et al., Joint entity and relation extraction based on a hybrid neural network, *Neurocomputing*, **257** (2017), 59–66. <https://doi.org/10.1016/j.neucom.2016.12.075>
35. K. Xue, Y. Zhou, Z. Ma, T. Ruan, H. Zhang, P. He, Fine-tuning BERT for joint entity and relation extraction in Chinese medical text, in *2019 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, (2019), 892–897. <https://dx.doi.org/10.1109/bibm47256.2019.8983370>
36. G. Bekoulis, J. Deleu, T. Demeester, C. Develder, Joint entity recognition and relation extraction as a multi-head selection problem, *Expert Syst. Appl.*, **114** (2018): 34–45. <https://dx.doi.org/10.1016/j.eswa.2018.07.032>
37. S. Zheng, F. Wang, H. Bao, Y. Hao, P. Zhou, B. Xu, et al., Joint extraction of entities and relations based on a novel tagging scheme, in *Proceedings of the Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL)*, (2017), 1227–1236. <https://dx.doi.org/10.18653/v1/P17-1113>
38. A. Katiyar, C. Cardie, 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 (ACL)*, (2017), 917–928. <https://dx.doi.org/10.18653/v1/P17-1085>
39. X. Li, F. Yin, Z. Sun, X. Li, A. Yuan, D. Chai, et al., Entity-relation extraction as multi-turn question answering, in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*, (2019). <https://dx.doi.org/10.18653/v1/P19-1129>
40. D. Dai, X. Xiao, Y. Lyu, S. Dou, Q. She, H. Wang, Joint extraction of entities and overlapping relations using position-attentive sequence labeling, in *Proceedings of the AAAI Conference on Artificial Intelligence*, **33** (2019), 6300–6308. <https://doi.org/10.1609/aaai.v33i01.33016300>
41. T. J. Fu, P. H. Li, W. Y. Ma, 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 (ACL)*, (2019), 1409–1418. <https://doi.org/10.18653/v1/P19-1136>
42. M. Eberts, A. Ulges, Span-based joint entity and relation extraction with transformer pre-training, *arXiv preprint*, (2019), arXiv:1909.07755. <https://doi.org/10.48550/arXiv.1909.07755>
43. Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, R. Soricut, ALBERT: A lite BERT for self-supervised learning of language representations, *arXiv preprint*, (2019), arXiv:1909.11942. <https://doi.org/10.48550/arXiv.1909.11942>

44. T. Mikolov, K. Chen, G. Corrado, J. Dean, Efficient estimation of word representations in vector space, *arXiv preprint*, (2013), arXiv:1301.3781. <https://doi.org/10.48550/arXiv.1301.3781>
45. Y. Kim, Convolutional neural networks for sentence classification, in *Proceedings of the Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, (2014), 1746–1751. <https://doi.org/10.3115/v1/D14-1181>
46. Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, et al., RoBERTa: A robustly optimized BERT pretraining approach, *arXiv preprint*, (2019), arXiv:1907.11692. <https://doi.org/10.48550/arXiv.1907.11692>
47. J. Devlin, M. W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, *arXiv preprint*, (2019), arXiv:1810.04805. <https://doi.org/10.48550/arXiv.1810.04805>
48. A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. Gomez, et al., Attention is all you need, *arXiv preprint*, (2017), arXiv:1706.03762. <https://doi.org/10.48550/arXiv.1706.03762>
49. H. Yan, B. Deng, X. Li, X. Qiu, TENER: Adapting transformer encoder for named entity recognition, *arXiv preprint*, (2019), arXiv:1911.04474. <https://doi.org/10.48550/arXiv.1911.04474>
50. D. Hendrycks, K. Gimpel, Gaussian Error Linear Units (GELUs), *arXiv preprint*, (2016), arXiv:1606.08415. <https://doi.org/10.48550/arXiv.1606.08415>
51. Y. Zhang, H. Zhao, B. Li, Semantic slot filling based on BERT and BiLSTM, *Comput. Sci.*, **48** (2021), 247–252. <https://doi.org/10.11896/jsjx.191200088>
52. J. Lafferty, A. McCallum, F. C. Pereira, Conditional random fields: Probabilistic models for segmenting and labeling sequence data, in *Proceedings of the Eighteenth International Conference on Machine Learning*, (2001), 282–289. <https://dl.acm.org/doi/10.5555/645530.655813>
53. D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, *arXiv preprint*, (2014), arXiv:1412.6980. <https://doi.org/10.48550/arXiv.1412.6980>
54. A. Viterbi, Error bounds for convolutional codes and an asymptotically optimum decoding algorithm, *IEEE Trans. Inf. Theory*, **13** (1967), 260–269. <https://doi.org/10.1109/TIT.1967.1054010>
55. E. Strubell, P. Verga, D. Belanger, A. McCallum, Fast and accurate entity recognition with iterated dilated convolutions, in *Proceedings of the Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, (2017), 2670–2680. <https://doi.org/10.18653/v1/D17-1283>
56. Z. Huang, W. Xu, K. Yu, Bidirectional LSTM-CRF models for sequence tagging, *arXiv preprint*, (2015), arXiv:1508.01991. <https://doi.org/10.48550/arXiv.1508.01991>
57. X. Jin, J. Zhang, J. Kong, T. Su, Y. Bai, A reversible automatic selection normalization (RASN) deep network for predicting in the smart agriculture system, *Agronomy*, **12** (2022), 591. <https://doi.org/10.3390/agronomy12030591>

58. B. Gupta, A. Gaurav, P. Panigrahi, V. Arya, Analysis of artificial intelligence-based technologies and approaches on sustainable entrepreneurship, *Technol. Forecasting Social Change*, **186** (2023), 122152. <https://doi.org/10.1016/j.techfore.2022.122152>



AIMS Press

©2023 the Author(s), licensee AIMS Press. This is an open access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0>)