



OPEN ACCESS

EDITED BY

Xuzhen Zhu,
Beijing University of Posts and
Telecommunications (BUPT), China

REVIEWED BY

Qiong Hu,
University of Colorado Denver,
United States
Fei Gao,
Shanghai Jiao Tong University, China
Xinning Wang,
Ocean University of China, China

*CORRESPONDENCE

Wenjing Yan,
yanwenjing@butbu.edu.cn

SPECIALTY SECTION

This article was submitted
to Social Physics,
a section of the journal
Frontiers in Physics

RECEIVED 15 September 2022

ACCEPTED 27 October 2022

PUBLISHED 15 November 2022

CITATION

Zhang Q, Wei S, Li Z and Yan W (2022),
Combining NSP and NER for public
opinion event extraction model.
Front. Phys. 10:1044919.
doi: 10.3389/fphy.2022.1044919

COPYRIGHT

© 2022 Zhang, Wei, Li and Yan. This is an
open-access article distributed under
the terms of the [Creative Commons
Attribution License \(CC BY\)](https://creativecommons.org/licenses/by/4.0/). The use,
distribution or reproduction in other
forums is permitted, provided the
original author(s) and the copyright
owner(s) are credited and that the
original publication in this journal is
cited, in accordance with accepted
academic practice. No use, distribution
or reproduction is permitted which does
not comply with these terms.

Combining NSP and NER for public opinion event extraction model

Qingchuan Zhang, Siwei Wei, Zihan Li and Wenjing Yan*

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

Event extraction in the field of public opinion aims to extract important event arguments and their corresponding roles from the moment-to-moment generated opinion reports. Most of the existing research methods divide the task into three subtasks: event trigger extraction, event type detection, and event argument extraction. Despite the remarkable achievements of the event argument extraction paradigm combining part-of-speech (POS) and event trigger features, the performance of POS features in combinatorial event argument extraction tasks is struggling due to its inherent semantic diversity in Chinese. In addition, previous research work ignored the deep semantic interaction between event trigger and text. To address the aforementioned problems, this paper proposes an opinion event extraction model (NN-EE) combining NSP and NER, which alleviates the lack of performance of combinatorial event argument extraction by introducing NER technology. Meanwhile, the event trigger features are incorporated into the NSP mechanism of the pre-trained language model BERT to prompt the model to learn the deep semantic interaction between the event trigger and original text. The results of the self-constructed food opinion report dataset (FD-OR) in this paper show that the NN-EE model achieves optimal performance.

KEYWORDS

event extraction, event argument extraction, NSP, NER, BERT

1 Introduction

With the rapid development of the Internet, more and more people choose to share what they see and hear around them through social media. In the face of massive data dissemination, quickly, accurately, and automatically extracting event types and their core arguments from the data has become an urgent problem in the field of public opinion event analysis. A key technology in the field of information extraction, event extraction has attracted extensive attention in recent years, aiming at identifying event types from unstructured texts and extracting important arguments with different roles from them manually or automatically. As shown in [Figure 1](#), for a given input text, the event extraction model first identifies the event trigger “Crack down” contained in the text and uses this as an important clue to judge the event type the text belongs to: “Counterfeit.” Then, using the previously identified trigger “Crack down” and the named entities



identified in the text as clues, the event argument and their corresponding roles of the event “Counterfeit” is extracted from the text by the event argument extraction model, that is, {“Place of occurrence”: “Dongxing,” “Counterfeit food”: “G7coffee,” “Counterfeit brand”: “G7”}.

To automatically extract the core event arguments and their corresponding roles from the text, most of the early research methods used the pipeline-based event extraction framework [1–4]. Under this framework, the task was divided into three subtasks, event trigger extraction, event type extraction, and event argument extraction. Although the pipeline-based framework has good flexibility, the results of event argument extraction tasks are completely dependent on the performance of the event trigger extraction task. At the same time, because the execution of the three subtasks is completely independent, the error propagation between models is irreversible, and the model has a serious error accumulation problem.

In view of the aforementioned problems of the pipeline framework, in recent years, a large number of end-to-end joint event extraction frameworks have been proposed and have achieved excellent performance [5–9]. The extraction of event triggers and event arguments can be performed simultaneously by the joint framework. Although this framework can avoid the error accumulation problem, to the best of our knowledge, none of the joint frameworks make good use of semantic dependencies and ignore the relationship between event triggers and event arguments.

Currently, neural networks are widely used in the research of event extraction tasks, and by introducing pre-trained models, syntactic features, and semantic features, the performance of neural network models has been continuously improved, and they even perform well on multilingual event extraction tasks [11–15]. However, the POS feature, a grammatical feature commonly used in event extraction, does not perform well in the task of extracting combined event arguments in the Chinese text [3].

In this paper, we propose a novel approach to use the triggers from the already well-performing event-triggered word extraction task with the original text as input to the NSP task by the NSP mechanism, thus prompting the model to learn the deep semantic interactions between the event trigger and original text. In addition, we also introduce the named entity recognition (NER) technology to solve the problem that the model does not perform well for combinatorial event argument extraction in Chinese. In turn, an event extraction framework combining Next Sentence Predict (NSP) and NER is developed, which solves the difficulty that the commonly used POS features make it difficult for the model to accurately identify Chinese combinatorial event arguments and deep semantic interaction between event trigger features and text. Extensive comparative experiments were conducted on the self-constructed dataset, and the results show that the NN-EE model proposed in this paper achieves optimal performance, proving the effectiveness of the event extraction model combining the NSP mechanism with the NER technique, and the model performance is improved.

The main contributions of this paper are summarized as follows:

- (1) In this paper, we proposed an event argument extraction model combining the NSP mechanism to deepen the deep semantic interaction between event triggers and text by introducing the NSP mechanism.
- (2) Unlike previous research methods that only use POS features to stitch with word-embedding vectors, this paper stitches entity-type label-embedding vectors with word-embedding vectors, which can significantly improve the accuracy of the event argument extraction model in identifying Chinese combinatorial event arguments.
- (3) In this paper, we constructed a dataset for the food opinion reporting domain (FD-OR) with a complex syntactic structure and wide semantic coverage of the corpus, and our model achieves optimal performance on this dataset.

2 Related work

Previous research work can be divided into two frameworks according to different stages of task execution, one is the pipeline framework and the other is the joint framework.

2.1 Pipeline framework

DMCNN framework: Ref. [2] proposed a method based on a dynamic multi-pooling convolutional neural network (DMCNN), which uses dynamic multi-pooling layers according to event triggers and candidate arguments to capture sentence-level clues without using complex NLP tools and more comprehensively retain information. However, this method is not flexible enough to cope with the situation where there is only one event in a sentence but multiple occurrences of the same trigger, which often occurs in real texts.

QA-based framework: Proposed a method to transform event trigger extraction and event argument extraction into a question answering (QA) task to extract event arguments in an end-to-end manner. This method can extract event arguments that appear during training. Meanwhile, Ref. [4] proposed a method to transform the event extraction task into a multi-round question-and-answer task. Ref. [13] proposed a method to transform the event extraction task into a machine reading comprehension task. However, they all ignored the deep semantic interaction between event triggers and original text.

2.2 Joint framework

JMEE framework: Ref. [14] proposed a novel multiple-event extraction (JMEE) framework. By introducing synchronous shortcut arcs to enhance information flow and an attention-based graph convolution network to simulate graph information, multiple event triggers and parameters and their roles can be jointly extracted. However, the joint framework mentioned above makes good use of semantic dependencies and ignores the deep relationship between event triggers and event arguments.

Cas EE framework: Ref. [15] proposed a method to extract overlapping events by cascading decoding. This method carries out event type detection, event trigger extraction, and event parameter extraction in turn, in which all subtasks learn together in a framework, so that the dependency between subtasks can be captured, which successfully solves the overlapping problem in event extraction. Meanwhile, Ref. [16] proposed a deep learning model using a bidirectional recurrent neural network (RNN) to induce shared hidden representations of words in sentences for all three subtasks, and the method could improve the performance of event extraction using the interactions between subtasks. However, while all the aforementioned models learn deep relationships between event

triggers and text, no other semantic or syntactic features are used to improve the overall performance of the models.

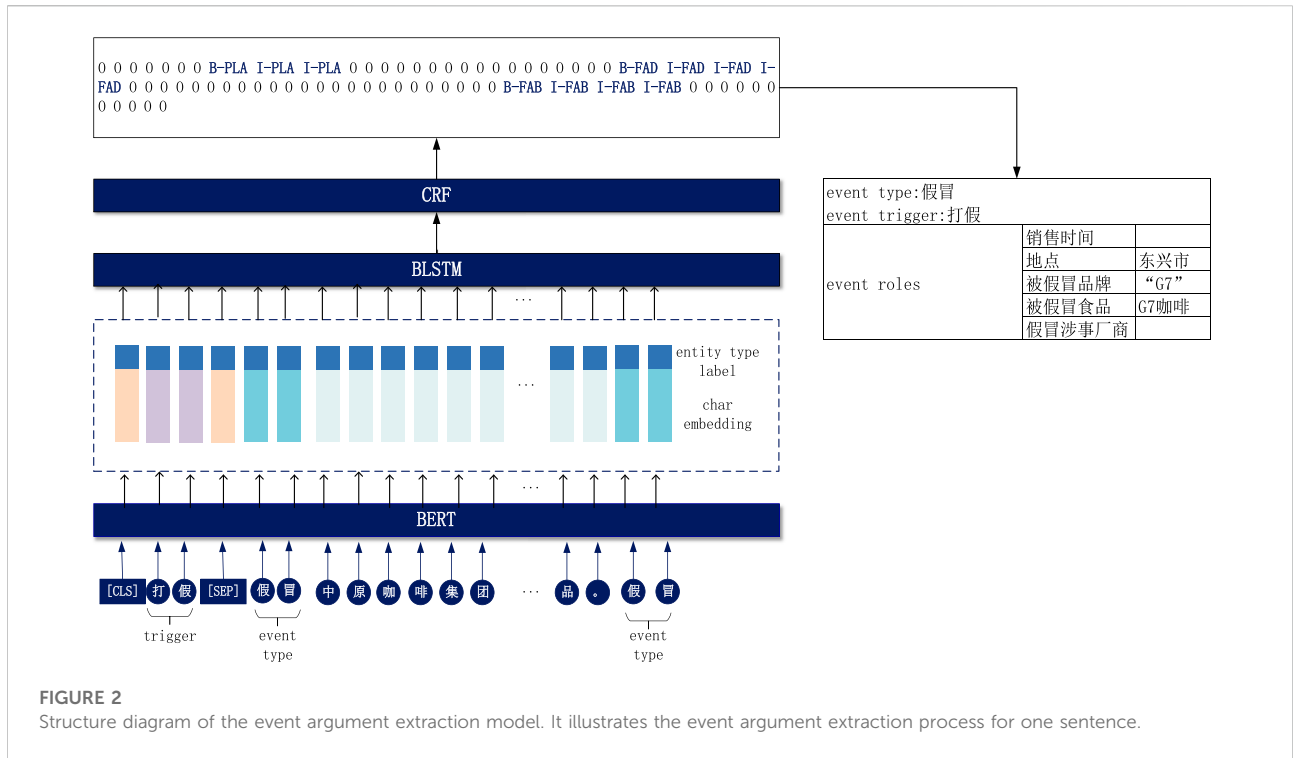
3 Methods

The purpose of EE is to extract event triggers and detect the event types, event arguments, and their corresponding roles contained in a sentence. In this paper, the event extraction task is modeled using the pipeline framework and is divided into three subtasks: event trigger extraction, event type detection, and event argument extraction.

In this paper, the event type detection task is considered a multi-label classification task and the event trigger extraction task as a NER task. This paper also uses the BERT-BLSTM model combining the sigmoid function as the event type detection model and the BERT-BLSTM-CRF model as the event trigger extraction model, which has been shown to perform excellently in previous NER tasks [17]. The models for both of these tasks already have F1 values above 99% on the FD-OR dataset constructed in this paper, and there is little room for further optimization. Therefore, in this study, we only optimize the event argument extraction models with poorer performance. This study mitigates the lack of performance of combinatorial event argument recognition in the event argument extraction task by introducing the NER technique; at the same time, by incorporating event triggers into the NSP mechanism of the pre-trained language model Bidirectional Encoder Representations from Transformer (BERT) [18], the event argument extraction model is motivated to learn the deep semantic interactions between the event trigger and the original text. The improvement in performance of the event argument extraction model is achieved by the aforementioned method.

Denoted by $S = \{c_1, c_2, c_3, \dots, c_n\}$, a sentence of length n , where c_i denotes the i th character; by $E = \{e_1, e_2, e_3, \dots, e_k\}$, the k entities in the sentence, where k is the total number of entities in the sentence; by $V = \{v_1, v_2, \dots, v_m\}$, the m events in the sentence, where m is the total number of events occurring in the sentence; by T the event trigger word corresponding to the m events in the sentence; and by using BIO annotation to label each token c_i with corresponding label y_i , to indicate the case where the event argument corresponding to the event type consists of one or more consecutive characters.

The NN-EE model proposed in this paper is an improvement on the BERT-BLSTM-CRF model, which consists of the following two main modules: (i) char encoder, which represents sentences with vectors and improves the performance of the model for combinatorial event argument extraction by introducing the entity type label feature; (ii) the event argument extraction decoder, which enhances the model's learning of deep semantic interactions between the event trigger and the original text by using event



triggers as the text1 of the NSP mechanism. We first obtained the encoding of the input text through the BERT layer and stitch it with the NER features, and finally got the current event type corresponding to the event arguments and its role through the BLSTM + CRF layer, as shown in Figure 2.

3.1 Char encoder

In the char encoder, the char embedding corresponding to each token c_i in the sentence is obtained by the BERT pre-training model. In order to solve the problem of unsatisfactory performance in the event extraction task for the recognition of combinatorial event arguments in Chinese, this paper splices the char embedding with the entity-type label-embedding vector, thus transforming c_i into a real-valued vector x_i .

- Character vector for c_i : When a character passes through the BERT model, character vectors $Z = \{z_1, z_2, z_3, \dots, z_n\}$ are generated, as shown in the following equation:

$$Z = \{z_1, z_2, z_3, \dots, z_n\} = \text{BERT encoder}(S) \quad (1)$$

where $S = \{c_1, c_2, \dots, c_n\}$, is a sentence of length n .

- Entity-type label feature for c_i : This is used to extract the required entity types by using the public NER model in the open-source Chinese natural language processing tool HanLP to identify the event arguments that occur in the sentence. The entities appearing in the sentence are then labeled with their

entity type at the position corresponding to the characters, and “O” at other positions. The entity type labels are then transformed by the embedding layer into a real-valued vector $B = \{b_1, b_2, b_3, \dots, b_n\}$.

Then, the method of obtaining the representation X of the token used for argument extraction is needed, as shown in the following equation:

$$X = [Z; B], \quad (2)$$

where $[-; -]$ denotes the splicing of the vectors.

The conversion x_i from token c_i to a real-valued vector is to convert the input sentence $S = \{c_1, c_2, c_3, \dots, c_n\}$ into a string of real-valued vectors $X = \{x_1, x_2, x_3, \dots, x_n\}$. The sentence vectors are then used as input to the next model in turn, so that the model learns a more efficient vector representation for the event argument extraction task.

3.2 Event argument extraction decoder

In order to enhance the connection between the preceding and following subtasks in the pipeline framework, this paper uses the output of the event trigger extraction task and the event type detection task as priori knowledge to play a role in the event argument extraction task. In this paper, we use the BIO annotation method to annotate each c_i in a sentence with its corresponding label y_i .

In this paper, the BLSTM-CRF model is used to complete the decoding of the event argument extraction task, and the aforementioned real-valued vector $X = \{x_1, x_2, x_3, \dots, x_n\}$ is the input of the BLSTM layer. After the BLSTM layer outputs the hidden layer vector $H = \{h_1, h_2, h_3, \dots, h_n\}$, the vector dimension is changed to the total number of label categories to be identified through the fully connected layer, thus obtaining the input $P = \{p_1, p_2, p_3, \dots, p_n\}$ of the CRF layer. The output of the fully connected layer can only obtain the relationship between $S = \{c_1, c_2, c_3, \dots, c_n\}$ and the tagged BIO labels, but lacks the consideration of the relationship between BIO labels, thus some invalid BIO labels are predicted. With the conditional random field (CRF) method, the predicted BIO tags can be constrained to reduce the probability of predicting invalid BIO tags, and the final output is a sequence of tags $L = \{l_1, l_2, l_3, \dots, l_n\}$ consisting of each token c_i corresponding to tag l_i predicted by the model. The loss function of the model is as shown in the following equation:

$$LOSS = \sum_{i=1}^n I(CRF(p_i), y_i), \quad (3)$$

where $I(CRF(p_i), y_i)$ is the negative log *likelihood* of the comparison between the predicted value $CRF(p_i)$ of the entity model and the true value y_i of the sequence labeling.

To improve the precision of the BLSTM-CRF model in extracting event arguments corresponding to event types, in this paper, event type names are spliced on both sides of the original sentence and encoded as part of the input sentence. By acting through the BLSTM layer, the event type names on both sides of the input sentence cause the hidden layer vectors corresponding to the characters in the original sentence to learn information about the event type, thus improving the overall performance of the model.

Because in the FD-OR constructed in this study, important event arguments often appear before and after the trigger (Figure 1), “Counterfeit food” appears at a position one character away from the trigger word, making deep semantic

interaction between the event trigger word and the original sentence, which is crucial to improve the performance of event argument extraction.

In order to solve the challenge, this paper introduces the approach of combining the NSP task in BERT to capture the relationship between the event trigger and the original sentence as mentioned in Ref. [10]. The NSP task in the BERT model is to predict whether two sentences are preceding or following sentences, and the similarity between two sentences text1 and text2 is calculated by the text similarity task; if they are similar, then it is considered that text2 is the next sentence of text1, and *vice versa*. In this process, the long-distance interdependent features in the sentences are captured by BERT’s multi-headed self-attention mechanism, which enables deep semantic interaction between text1 and text2 and captures the relationship between text1 and text2. The attention formula is:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK}{\sqrt{d_k}}\right)V, \quad (4)$$

where Q (query) is the query vector, K (key) is the queried vector, V (value) is the content vector, and d_k is the dimensionality of Q, K. In the self-attention mechanism, Q is x_i and K, V is $X = \{x_1, x_2, x_3, \dots, x_n\}$. Multi-headed self-attention is a mechanism that divides Q, K, and V into multiple branches to learn different features.

There are two special symbols [CLS] and [SEP] in BERT. [CLS] indicates that the feature is used in a classification model; for non-classification models, this symbol can be omitted. The [SEP] symbol is used to break the two sentences of text1 and text2 of the input. After splicing the event argument names and performing the input processing for the NSP task in this paper, the input to the model is shown in Figure 2.

The NSP task is applied to the proposed event argument extraction model by using the event trigger word as text1 in the NSP task and the original sentence as text2. The NSP task enables deep semantic interaction between the event trigger and the original sentence, thus improving the performance of the event argument extraction model. In addition, the entity type labels of the two parts mentioned previously are complemented with “O” for the corresponding characters.

4 Experimental results

4.1 Experimental dataset

In this paper, Chinese food opinion report data were constructed as the experimental dataset. The corpus of news reports in the experimental dataset was particularly sourced from China Quality News Network, which is under the supervision of the State Administration for Market Regulation of China, and the

TABLE 1 Pre-defined event schema of FD-OR.

Event type	Event role
Counterfeit (fake)	Sales time (sell time), (place), counterfeit brand (be faked brand), counterfeit food (be faked food), and fake the manufacturer involved (fake factory)
Exceed the standard (exceeded)	Sales time (sell time), (place), counterfeit brand (over standard food), counterfeit food (over standard item), and manufacturers involved in exceeding the standard (over standard factory)
Presence of foreign body (have foreign substance)	Sales time (sell time), (place), presence of foreign body food (food with foreign substance), foreign body present (foreign substance), and manufacturers involved in foreign bodies (foreign substance-related factory)

TABLE 2 Experimental dataset.

Dataset		Training	Validation	Test	Event type
FD-OR	Number of sentences	1800	257	514	3
	Avg event argument	3	2	3	-

Chinese mainstream news platforms Baidu News and Today's Headlines. The dataset is available at <http://180.76.244.155:8080/FD-OR.zip>. Three event types were included in the dataset. Different event types corresponding to different event arguments are as shown in Table 1.

In the ratio of 7:2:1, the experimental dataset is divided into three parts: training set, testing set, and validation set. The specific dataset division is shown in Table 2.

Another dataset is the food safety news reports dataset (FD-SR). Five event types were included in the dataset. Different event types corresponding to different event arguments are shown in Table 3.

4.2 Evaluation standard setting

In this paper, the experimental results of precision, recall, and F1 score are used as model performance measures.

The formula for calculating the precision is:

$$P = \frac{TP}{TP + FP} \quad (5)$$

where TP indicates the number of classes that are themselves positive and that the model correctly predicts as positive, and FP indicates the number of classes that are themselves negative and that the model predicts as positive.

The formula for calculating the recall is:

$$R = \frac{TP}{TP + FN} \quad (6)$$

where TP is the same as the aforementioned formula and FN indicates the number of classes that are themselves positive but that the model predicts as negative.

The formula for calculating the F1 score is:

$$F1 = \frac{P * R * 2}{P + R} \quad (7)$$

The FD-OR constructed in this paper is a balanced dataset, and precision and recall are two contradictory metrics. The F1 score is essentially the summed average of precision and recall, so in order to better evaluate the performance of the model, the F1 value is used as the overall evaluation metric to balance precision and recall, and the loss function during the training process is shown in Figure 3.

TABLE 3 Pre-defined event schema of FD-SR.

Event type	Event role
Food sampling announcement	(food name), (time), (testing unit), and (units involved)
Malignant emergencies	(time), (position), (consequences), (number of people affected), and (units involved)
Food quality defects	(food name), (manufacturer), (quality defects), and (food purchase source)
Production environment exposure	(time), (place), (unit involved), (type of production environment), and (exposure issues)
Off-line dining complaints	(time), (place), (unit involved), (complaint food), and (complaint type)

4.3 Experimental parameter settings

In terms of experimental parameter settings, the main parameter information of the model in this paper is finalized by experience, continuous experimentation and adjustment, and corpus specifics. The hidden layer of the BERT model has 12 layers and its output vector dimension is 768; the hidden layer of the BLSTM model has an output vector dimension of 256, and the dimension of the entity type label vector is 80. One of the open-source deep learning frameworks, PyTorch (<https://pytorch.org/>), was used to build the experimental platform to develop deep learning models. In the experiments, the values of the main parameters of the proposed model in this paper are as shown in Table 4. The hyperparameters in the experiments were determined by the specifics of the experiments and the corpus. Following the empirical value, we set the output vector dimension of the BLSTM model as 256, when the parameter is set to dimensions of entity-type label vector as 80, epochs as 40, max length as 150, and learning rate as 1e-5, the model obtains the optimal performance.

4.4 Experimental results and analysis

Through the following experiments, the NN-EE model proposed in this paper is compared with the following neural network-based event argument extraction models.

- 1) Baseline: This model is the BLSTM-CRF model.
- 2) BERT-BLSTM-CRF: This model is the BERT + Baseline model.

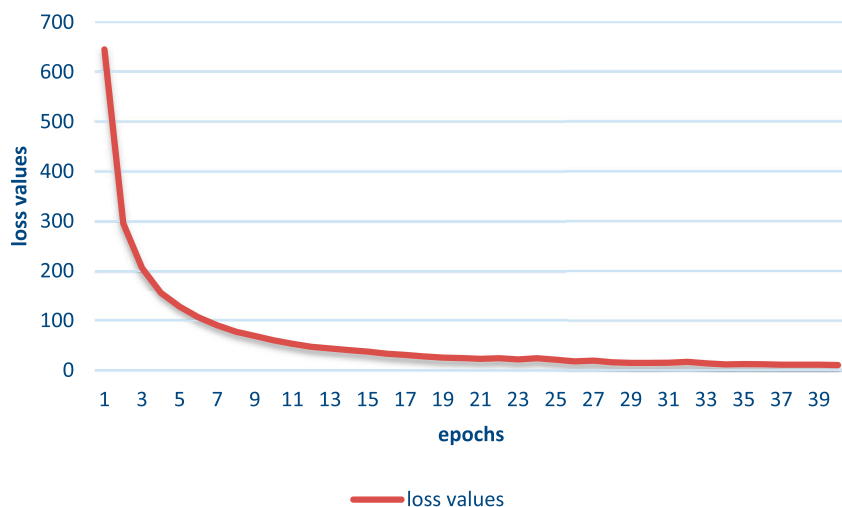


FIGURE 3
Loss function during the training process.

- 3) BERT–BLSTM–CRF + NER: This model is based on the BERT + Baseline model, where all entities appearing in the sentence are annotated, and then the entity type labels are encoded to be spliced with the output of the BERT layer, which is then input to the BLSTM layer.
- 4) Baseline + NSP: The model is based on the BERT + Baseline model and introduces the NSP mechanism from the BERT model. In this mechanism, the event trigger is used as text1 for this task, the symbol “ [SEP],” and the original sentence are spliced together and used as model input.
- 5) NN-EE: The model is a combination of the BERT + Baseline + NER model and the BERT–BLSTM–CRF + NSP model described previously.

Table 5 shows the overall performance of the aforementioned models on the FD-OR, a self-constructed dataset in this paper. In addition, to demonstrate the stability of our proposed model, we conducted comparative experiments on another dataset of food safety news reports (FD-SR). From the results, we can observe that our proposed model combining NSP and NER for public opinion event extraction (NN-EE) achieves the best F1 score across all the compared methods. NN-EE improves the F1 score to 96.83%, which is an improvement of 1.11% compared to the baseline model. It can be observed from experiments 1 and 2 that the recall of the model increased and the precision decreased, and from experiments 1 and 3 that the recall, precision, and F1 score of the model decreased when combining the NSP mechanism, and from experiments 1 and 4 that the precision, recall, and F1 score of the model increased. This shows that using either the NSP mechanism or the NER technique model

performance alone will decrease the performance of the model. The reason is that the NSP mechanism inherent in the BERT model can capture the semantic similarity between text_a and text_b. In this paper, the NSP mechanism can prompt the model to focus on the semantics of different parts of the text in the face of different trigger words and event types, alleviating the semantic interaction between the three that has been ignored in previous research work and improving the NER model’s targeted extraction of relevant entities’ capability of the NER model. Our NN-EE achieves optimal performance and demonstrates the effectiveness of combining the NSP mechanism with the NER model.

A specific event extraction example is presented to further explain our model, as shown in Figure 4. We first obtained the

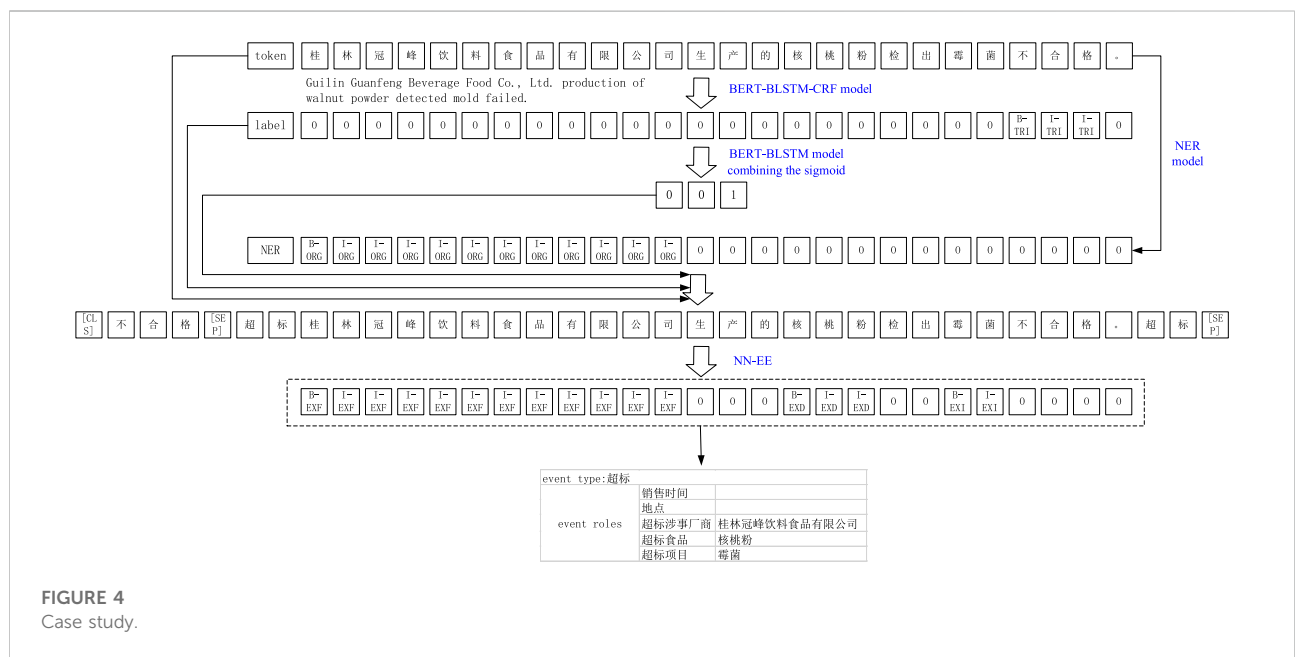
TABLE 4 Experimental parameter setting.

Parameter	Value
Epochs	40
Batch size	8
Dimensions of entity-type label vector	80
Maximum length	150
Learning rate	1e-5
Layer of BERT	12
Dimensions of BERT	768
Function of BERT	Tanh
Layer of BLSTM	2 layers
Dimensions of BLSTM	256
Function of BLSTM	ReLU tanh

TABLE 5 Experimental results of event extraction.

Model	FD-OR data			FD-SR data		
	P (%)	R (%)	F1 (%)	P (%)	R (%)	F1 (%)
Baseline	92.64	83.22	87.66	79.32	74.21	76.67
BERT + Baseline	97.11	94.37	95.72	85.12	84.77	84.94
BERT + Baseline + NER	97.82	95.51	96.65	88.47	84.33	86.35
BERT + Baseline + NSP	96.93	95.22	96.36	92.64	83.22	87.66
NN-EE (ours)	98.19	95.51	96.83	89.22	86.75	87.96

P, R, F1 have been mentioned in the section Evaluation standard setting. The explanation of FD-OR data, FD-SR data has been given in section Experimental dataset.



initial input event trigger word: 不合格 (failure) by the BERT-BLSTM-CRF trigger word extraction model, while regretting the use of the BERT-BLSTM event type detection model combined with the sigmoid function to obtain the event type: exceedance, and also the initial input sequence of named entities $N = \{O, O, O, \dots, B - TRI, I - TRI, I - TRI, O\}$. Then, we formed the token sequence $S = \{[CLS], hit, fake, [SEP], fake, w_1, \dots, w_n, fake, [SEP]\}$ by splicing the event trigger, event type, and input text according to the input format of the BERT model and obtain the event arguments and their roles in it using our proposed NN-EE model. For example, {manufacturer involved in exceeding the standard: Guilin Guanfeng Beverage and Food Co., Ltd., exceeding the standard food: walnut powder, exceeding the standard item: mold}. Finally, we combined the obtained entity results with the trigger words and event types obtained from the upstream task to complete the event extraction for this text.

5 Conclusion and discussion

In this paper, we propose an opinion event extraction model (NN-EE) combining NSP and NER, which solves the problem of insufficient performance in recognizing combinatorial event arguments in Chinese text by introducing NER technology. At the same time, by incorporating event trigger word features into the NSP mechanism of the pre-trained language model BERT, the model is motivated to learn the deep semantic interactions between the event trigger and original text. The model achieves optimal performance on FD-OR, a self-constructed food opinion reporting dataset in this paper, and the experiments demonstrate the effectiveness of the model. In the future, we will optimize the performance of the model in handling event arguments that are nested entities.

Data availability statement

The datasets presented in this study can be found in online repositories. The names of the repository/repositories and accession number(s) can be found in the article/Supplementary Material.

Author contributions

QZ contributed significantly to theoretical analysis and manuscript preparation. SW performed the experiments, and contributed to the conception of the study and model formulation. ZL provided many promising insights during the revision process. WY performed the data analyses and helped perform the analysis with constructive discussions. All authors contributed to the article and approved the submitted version.

Funding

This research was supported by Beijing Natural Science Foundation (grant no. 4202014), the National Key Technology R and D Program of China (grant no. 2021YFD2100605), the Natural Science Foundation of China (grant nos. 62006008 and 61873027), Humanity and Social

Science Youth Foundation of Ministry of Education of China (grant no. 20YJCZH229), Open Project Program of National Engineering Laboratory of Agri-Product Quality Traceability (grant no. AQT-2020-YB6), the Social Science Research Common Program of Beijing Municipal Commission of Education (grant no. SM202010011013), and Research Foundation for Youth Scholars of Beijing Technology and Business University (grant no. QNJJ2020-28).

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors, and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

- Zhang Z, Xu W, Chen Q. Joint event extraction based on skip-window convolutional neural networks. In: *Natural Language Understanding and Intelligent Applications* (2016). p. 324–34.
- Ji H, Grishman R. Refining event extraction through cross-document inference. In: *Proceedings of ACL-08: HLT* (2008). p. 254–62.
- Hong Y, Zhang J, Ma B, Yao J, Zhou G, Zhu Q. Using crossentity inference to improve event extraction. In: *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies* (2011). p. 1127–36.
- Ritter A, Etzioni O, Clark S. Open domain event extraction from twitter. In: *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (2012). p. 1104–12.
- Huang L, Cassidy T, Feng X, Ji H, Voss C, Han J, Sil A. Liberal event extraction and event schema induction. In: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics* (2016). p. 258–68.
- Zhang T, Ji H, Sil A. Joint entity and event extraction with generative adversarial imitation learning. *Data Intell* (2019) 1(2):99–120. doi:10.1162/dint_a_00014
- Chen Y, Xu L, Liu K, Zeng D, Zhao J. Event extraction via dynamic multi-pooling 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). p. 167–76.
- Nguyen TH, Cho K, Grishman R. Joint event extraction via recurrent neural networks. In: *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (2016). p. 300–9.
- Wang H, Zhu T, Wang M, Zhang G, Chen W. A prior information enhanced extraction framework for document-level financial event extraction. *Data Intelligence* (2021) 3(3):460–76. doi:10.1162/dint_a_00103
- Kan Z, Qiao L, Yang S, Liu F, Huang F. Event arguments extraction via dilate gated convolutional neural network with enhanced local features. *IEEE Access* (2020) 8:123483–91. doi:10.1109/access.2020.3004378
- Du X, Cardie C. Event extraction by answering (almost) natural questions. In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics (2020).
- Li F, Peng W, Chen Y, Wang Q, Pan L, Lyu Y, Zhu Y. Event extraction as multi-turn question answering. In: *Findings of the Association for Computational Linguistics: EMNLP 2020* (2020). p. 829–38.
- Liu J, Chen Y, Liu K, Bi W, Liu X. Event extraction as machine reading comprehension. In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (2020). p. 1641–51.
- Liu X, Luo Z, Huang H. Jointly multiple events extraction via attention-based graph information aggregation. In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Brussels, Belgium, October 31–November 4, 2018 (2018). p. 1247–1256. Available at: <https://aclanthology.info/papers/D18-1156/d18-1156>.
- Sheng J, Guo S, Yu B, Li Q, Hei Y, Wang L, et al. CasEE: A joint learning framework with cascade decoding for overlapping event extraction. In: *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, Association for Computational Linguistics, Online (2021). p. 164–174. Available at: <https://aclanthology.org/2021.findings-acl.14>
- Nguyen TM, Nguyen TH. One for all: Neural joint modeling of entities and events. *Proc AAAI Conf Artif Intelligence* (2019) 33:6851–8. doi:10.1609/aaai.v33i01.33016851
- Luo Y, Xiao F, Zhao H. Hierarchical contextualized representation for named entity recognition. *Proc AAAI Conf Artif intelligence* (2020) 34:8441–8. doi:10.1609/aaai.v34i05.6363
- Devlin J, Chang M-W, Lee K, Toutanova K. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018).