Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2023.040751

ARTICLE

# Multi-Modal Military Event Extraction Based on Knowledge Fusion

**Yuyuan Xiang, Yangli Jia*, Xiangliang Zhang and Zhenling Zhang**

School of Computer Science, Liaocheng University, Liaocheng, 252059, China

*Corresponding Author: Yangli Jia. Email: jiayangli@lcu.edu.cn

Received: 29 March 2023 Accepted: 29 August 2023 Published: 31 October 2023

**ABSTRACT**

Event extraction stands as a significant endeavor within the realm of information extraction, aspiring to automatically extract structured event information from vast volumes of unstructured text. Extracting event elements from multi-modal data remains a challenging task due to the presence of a large number of images and overlapping event elements in the data. Although researchers have proposed various methods to accomplish this task, most existing event extraction models cannot address these challenges because they are only applicable to text scenarios. To solve the above issues, this paper proposes a multi-modal event extraction method based on knowledge fusion. Specifically, for event-type recognition, we use a meticulous pipeline approach that integrates multiple pre-trained models. This approach enables a more comprehensive capture of the multidimensional event semantic features present in military texts, thereby enhancing the interconnectedness of information between trigger words and events. For event element extraction, we propose a method for constructing a priori templates that combine event types with corresponding trigger words. This approach facilitates the acquisition of fine-grained input samples containing event trigger words, thus enabling the model to understand the semantic relationships between elements in greater depth. Furthermore, a fusion method for spatial mapping of textual event elements and image elements is proposed to reduce the category number overload and effectively achieve multi-modal knowledge fusion. The experimental results based on the CCKS 2022 dataset show that our method has achieved competitive results, with a comprehensive evaluation value F1-score of 53.4% for the model. These results validate the effectiveness of our method in extracting event elements from multi-modal data.

**KEYWORDS**

Event extraction; multi-modal; knowledge fusion; pre-trained models

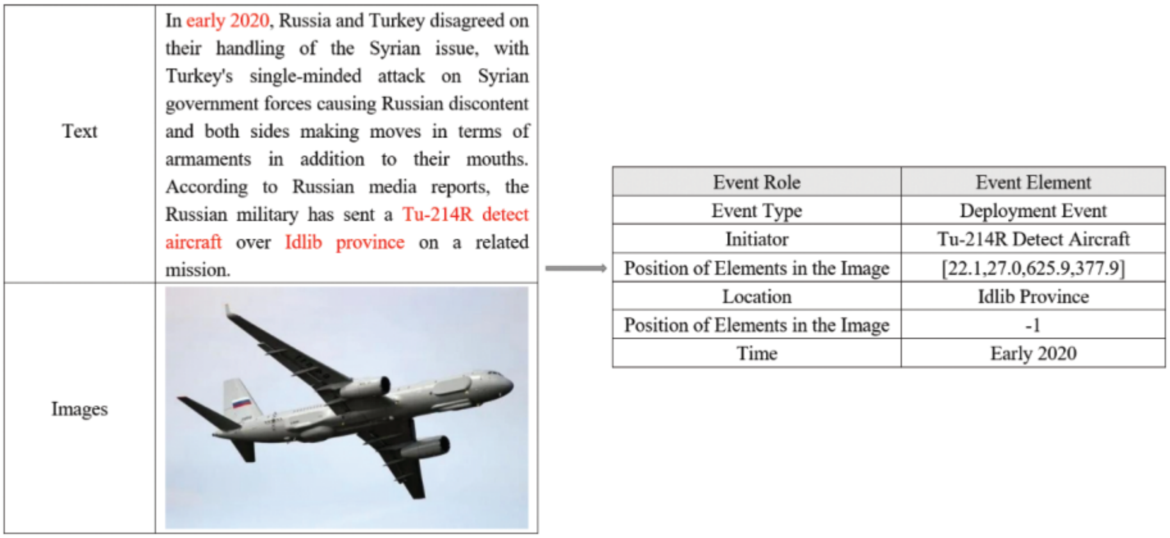
## 1 Introduction

Military informatization is the focus of modern military development. The application of event extraction technology in the military sector holds great potential for enhancing the efficiency of information acquisition. This technology enables the dynamic, real-time expansion of the information base and contributes to the effective management and analysis of military information. In recent years, internet-based equipment data has experienced significant growth. This kind of data is typically disseminated in the form of text, images, and other multi-modal content [1]. Military equipment data has gradually become an important resource and the basis for equipment requirement justification.

However, the currently available public datasets for multi-modal event extraction in the military domain are relatively limited. They suffer from a lack of diversity in data samples, exhibit a wide distribution of event elements, and pose challenges in effectively extracting crucial knowledge. Therefore, extracting relevant event types and elements from multi-modal military equipment data is of utmost importance. It facilitates the discovery of knowledge and application patterns that are suitable for equipment requirement argumentation.

Event extraction is a widely studied topic in natural language processing research [2,3]. Its primary objective is to automatically extract user-desired events from unstructured event information and represent them in a structured format. Event extraction techniques have a wide range of applications in the fields such as biomedical [4,5], judicial [6,7], social [8,9], journalistic [10,11], etc. But in the military domain, there is still a lack of effective event extraction approaches due to less research on military event extraction [12].

Usually, a military equipment event consists of triggers and arguments. Each trigger corresponds to a military equipment event and determines the corresponding event type. Arguments refer to multiple elements of the events. As shown in Fig. 1, the example consists of military equipment text and its corresponding image. In this example, we extracted information about the event type, the argument element, and the corresponding coordinate position of the event body in the image. If the object frame corresponding to the text is not detected in the image, it is marked as “−1”. However, most event extraction approaches are aimed at extracting arguments from the sentences of a document, such as the Knowledge Base Population (KBP) dataset[1], a popular event extraction dataset.



| | |
|---|---|
| Text | In early 2020, Russia and Turkey disagreed on their handling of the Syrian issue, with Turkey's single-minded attack on Syrian government forces causing Russian discontent and both sides making moves in terms of armaments in addition to their mouths. According to Russian media reports, the Russian military has sent a Tu-214R detect aircraft over Idlib province on a related mission. |
| Images | |

| Event Role | Event Element |
|---|---|
| Event Type | Deployment Event |
| Initiator | Tu-214R Detect Aircraft |
| Position of Elements in the Image | [22.1,27.0,625.9,377.9] |
| Location | Idlib Province |
| Position of Elements in the Image | -1 |
| Time | Early 2020 |

**Figure 1:** An example of a multi-modal event element

In this paper, we propose a multi-modal event extraction method based on knowledge fusion to address the challenge of event arguments for multi-modal data. The method consists of three subtasks: event type recognition, event argument extraction, and multi-modal knowledge fusion. In event type recognition, we use an event multi-label classification model and a trigger word extraction model to jointly define event types. The event multi-label classification model is built with BERT [13] as the baseline to obtain the semantic features and contextual information of the text. The trigger word

[1] https://tac.nist.gov/2017/KBP/

extraction model is built with ERNIE [14] as the baseline to obtain richer semantic information and distinguish ambiguity. In event argument extraction, we obtain a dynamic word vector representation based on contextual information from ERNIE. This representation captures bidirectional semantic information using a Bidirectional Gated Recurrent Unit (BIGRU) [15]. Then, Conditional Random Field (CRF) [16] decoding is used to identify event arguments. In multi-modal knowledge fusion, we use the BERT model to recognize argument entities and the YOLOv5 [17] model for target detection.

In addition, there is a shortage of sufficient annotated data and a significant presence of overlapping event arguments in the military domain. To train and evaluate our model, we use a data augmentation approach based on full-domain random substitution of parameter entities. This approach allows us to implement event expansion while maintaining syntactic and semantic invariance. We then construct an a priori template by integrating the model output used for event type recognition. Moreover, we have designed a fusion method for the spatial mapping of textual event elements and image elements. This method aims to reduce category count overload and effectively achieve multi-modal knowledge fusion. The contributions of this paper are summarized as follows:

- We propose a multi-modal military event extraction framework based on knowledge fusion. In this framework, text event elements and image elements are both mapped to the same label space, effectively integrating multi-modal knowledge.
- We propose a method to construct an a priori template of event types + trigger words based on the recognized event types. By effectively modeling the multidimensional semantics of overlapping parameters of different event types, more meaningful representations of semantic relationships between event elements can be learned.
- We conduct extensive experiments on the CCKS 2022 dataset[2] and demonstrated the effectiveness of the proposed method in multi-modal military event element extraction.

The remainder of the paper is structured as follows: Section 2 discusses the related work. In Section 3, we provide an overview of multi-modal event element extraction approaches. We first outline the general framework and then elaborate on event type recognition, event element extraction, and multi-modal knowledge fusion. Section 4 provides details about the experiment results and a discussion of the proposed methods. Finally, Section 5 concludes this paper with an overall summary and future works.

## 2 Related Works

Our research includes three objectives: event extraction approaches, object detection approaches, and multi-modal knowledge fusion. We review the major literature in the three areas.

### *2.1 Event Extraction*

Event extraction methods can be mainly divided into pattern-matching-based and machine-learning-based methods. Early event extraction usually uses pattern-matching-based methods. Riloff [18] mentioned the inclusion of event elements in the context of event trigger words by manually constructing a domain-specific dictionary for event extraction. However, the pattern-matching method depends on the specific form of domain-specific text and is less generalizable to the system.

In recent years, machine learning methods have gradually become the mainstream approach for event extraction. Compared with pattern-matching-based methods, machine learning methods are

[2] https://www.biendata.xyz/competition/KYDMTJSTP/

more adaptable to different domains and have better portability. Deep learning has become a very popular machine learning method and is widely used in event extraction tasks [19]. The first deep learning-based event extraction method utilized a pipeline-based model. Chen et al. [20] enhanced the traditional convolutional neural network model through a dynamic multi-pool mechanism and proposed a dynamic multi-pool convolutional neural network (DMCNN). This approach performs event extraction in two stages. To compensate for the shortcomings of the pipeline model, Tian et al. [21] employed a pre-trained language model for event extraction. They transformed the joint extraction task into an annotation problem and utilized an end-to-end model to extract entities and events. Lyu et al. [22] proposed a transformation-based neural network model that exploits the connection between the entity and event structures to perform joint entity and event extraction.

Although various event extraction methods have been proposed, they still produce unsatisfactory performance due to the complexity of military texts and the universality of overlapping event elements. Therefore, we propose a method to construct an a priori template of event types + trigger words based on the recognized event types. Our method can comprehensively capture the inherent multidimensional semantic features in military texts. At the same time, it can fully utilize the detailed features of trigger words, thereby promoting a deeper understanding of the semantic relationships between elements.

### 2.2 *Object Detection*

Object detection algorithms can be mainly classified into traditional object detection algorithms and deep learning-based object detection algorithms. Traditional object detection algorithms usually extract features manually. Felzenszwalb et al. [23] proposed a deformable part model for object detection. The model combines a Histogram of Oriented Gradient (HOG) and a Support Vector Machine (SVM) classifier. However, traditional object detection methods can only extract low-level image features and have low performance.

In recent years, most object detection methods have been based on deep learning. There are two main types of mainstream deep learning object detection algorithms: two-stage object detection algorithms and one-stage object detection algorithms. Two-stage detection algorithms first generate candidate regions and then classify the candidate regions. Girshick et al. [24] proposed the regions with CNN features (R-CNN) algorithm. The algorithm consists of generating candidate regions for region-based feature extraction, using Support Vector Machines (SVM) to detect the candidate regions, and determining their corresponding object classes and locations. The one-stage detection algorithm is an end-to-end object detection algorithm that accomplishes both object edge prediction and object classification. The YOLOv1 algorithm proposed by Redmon et al. [25] divides the image into many grids, and then localizes and classifies each grid of the image.

Although various object detection methods exist, they tend to share common limitations. These limitations include slow processing speeds, inefficient resource utilization, and challenges in generalizing to new object classes that are significantly different from the training dataset. To address these issues, our study adopted the YOLOv5 model to improve processing speed and resource efficiency. In addition, we constructed a target detection and recognition dataset using a combined human-machine label transformation approach, which effectively improves the overall performance of the model.

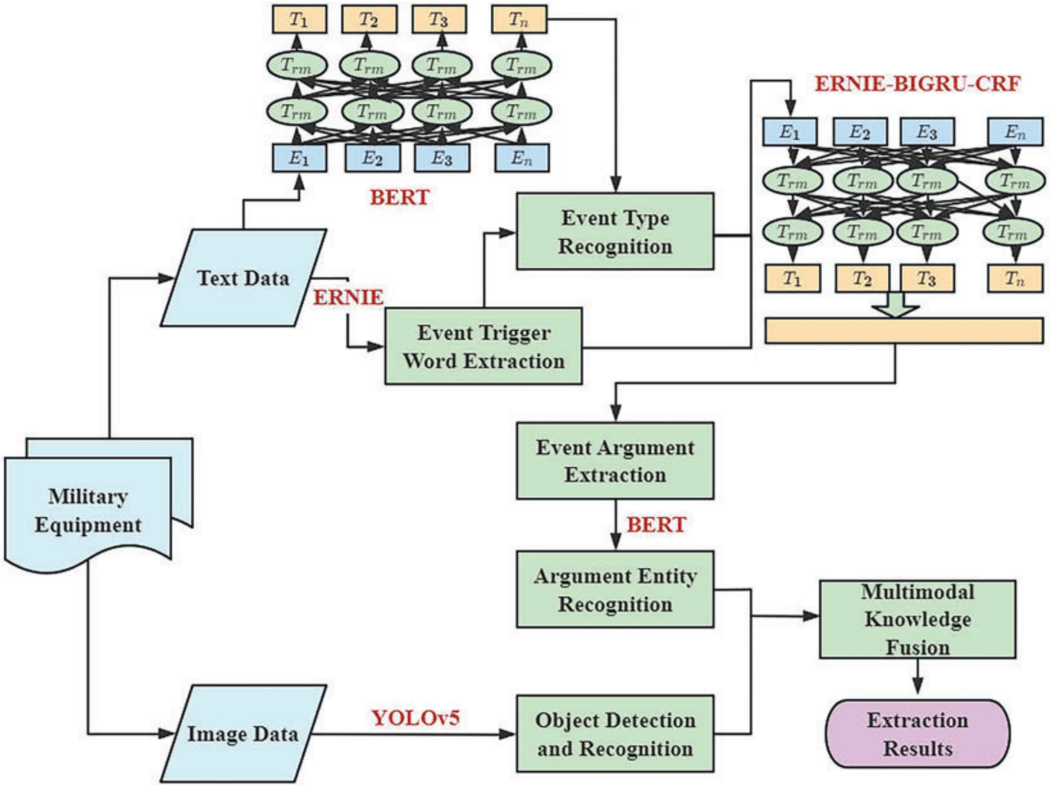
### 2.3 *Multi-Modal Knowledge Fusion*

Multi-modal knowledge fusion usually extracts feature representations of different modal information to achieve a collaborative representation of multi-modal data. Zhang et al. [26] proposed a multi-modal data source fusion model that utilizes gated cyclic units to capture the diversity of data sources bi-directionally. Additionally, they employed a hierarchical attention network to obtain a holistic representation of the information. Ding et al. [27] first extracted visually relevant multi-modal knowledge and then represented the multi-modal knowledge through a fine-grained explicit triad.

The majority of existing event extraction models predominantly concentrate on text-based scenarios, overlooking the potential of event element extraction from multi-modal data. As a result, the research on extracting event elements from multi-modal sources has received limited attention, leading to a relatively underexplored area of study. To effectively achieve multi-modal knowledge fusion, we propose a novel multi-modal label mapping method. This method facilitates the mapping of independent variables extracted from textual data and objects extracted from images into a unified label space, thus enabling the effective fusion of textual and visual information.

## 3 Materials and Methods

Our research proposes a multi-modal event element extraction framework that enables the extraction of a wider range of event types and elements from large-scale multi-modal military news documents. As shown in Fig. 2, the proposed framework comprises four phases organized in a pipeline fashion. These phases encompass event type recognition, event argument extraction, object detection and recognition, and multi-modal knowledge fusion.



T1
T2
T3
Tn
Trm
E1
E2
E3
En
BERT
ERNIE-BIGRU-CRF
Text Data
ERNIE
Event Type Recognition
Event Trigger Word Extraction
Military Equipment
Event Argument Extraction
BERT
Argument Entity Recognition
Multimodal Knowledge Fusion
Image Data
YOLOv5
Object Detection and Recognition
Extraction Results

**Figure 2:** A multi-modal event extraction framework

In the first phase of event extraction, trigger words are discovered from event sentences, and an event trigger word is a keyword that reflects the occurrence of an event. Domain experts annotate trigger words for different types of events and then expand the trigger word library by word vector similarity. A BERT-based multi-label classification model and an ERNIE-based trigger word extraction model are used to recognize the types of events in military news.

In the second phase, we constructed an a priori template of event types + trigger words based on the recognized event types to solve the problem of overlapping arguments of different event types in event sentences. Then, the ERNIE-BIGRU-CRF model is used to implement argument slot filling to extract the corresponding event arguments. In the third phase, the BERT model is used to recognize argument entities, and the YOLOv5 object detection algorithm is used to recognize object bounding boxes. Finally, the object bounding box coordinates are mapped to the text argument by using the multi-modal label mapping method.

### *3.1 Event Type Recognition*

#### *3.1.1 Event Trigger Word Extraction*

In event extraction, the trigger word can characterize the event occurrence, and it is the most important feature word to decide the event type. However, an event can be represented by different styles of triggers. There is a correspondence between the event type and the trigger word. The event type can be identified based on the trigger word. For example, the news item "French Phantoms attacked Palmyra and Raqqa in Syria" means that an attack event occurred due to the trigger word "attack". Therefore, this study constructs a trigger lexicon by labeling the trigger words of different types of events by domain experts. However, event features are difficult to be covered as well as may filter some words that can act as trigger words by themselves. Therefore, this study uses the ERNIE-based trigger word model to fully extract the trigger word information in military news to expand the trigger word database. The trigger words for different event types are shown in Table 1, the left column of the table represents the event type, while the right column contains the corresponding trigger words.

**Table 1:** Military events taxonomy

| Event type | Trigger words |
|---|---|
| Attack events | Attack/strike/impact/assault/destroy/air strike/kill/destroy/hit/bomb/combat/eliminate |
| Scouting events | Scouting/monitoring/surveillance/listening/search/monitoring/detection/tracking/search/scanning/patrol |
| Safeguard events | Safeguard/protect/maintain/guard/protect/support/supply/provide/replenish/refuel/transport |
| Blocking events | Blocking/blockade/containment/encirclement |
| Deployment events | Deployment/service/integration/joining/inclusion/berthing/formation/docking |
| Defensive events | Defensive/lock capture/capture/protect/defend/prevent/avoid |
| Maneuvering events | Maneuvering/fly over sail through/go through/enter/sail into/sail towards/enter port/arrive/transfer |

#### *3.1.2 Event Multi-Label Classification*

In event type recognition, since text contains multiple event types, a sentence may belong to multiple event types. Therefore, we need to use a multi-label text classification algorithm [28] to identify event types. Since a text contains a large number of unlabeled events, we propose to add a multi-label classification model with empty event classes to perform event multi-label classification. Fig. 3 shows an overview of the multi-label classification model.

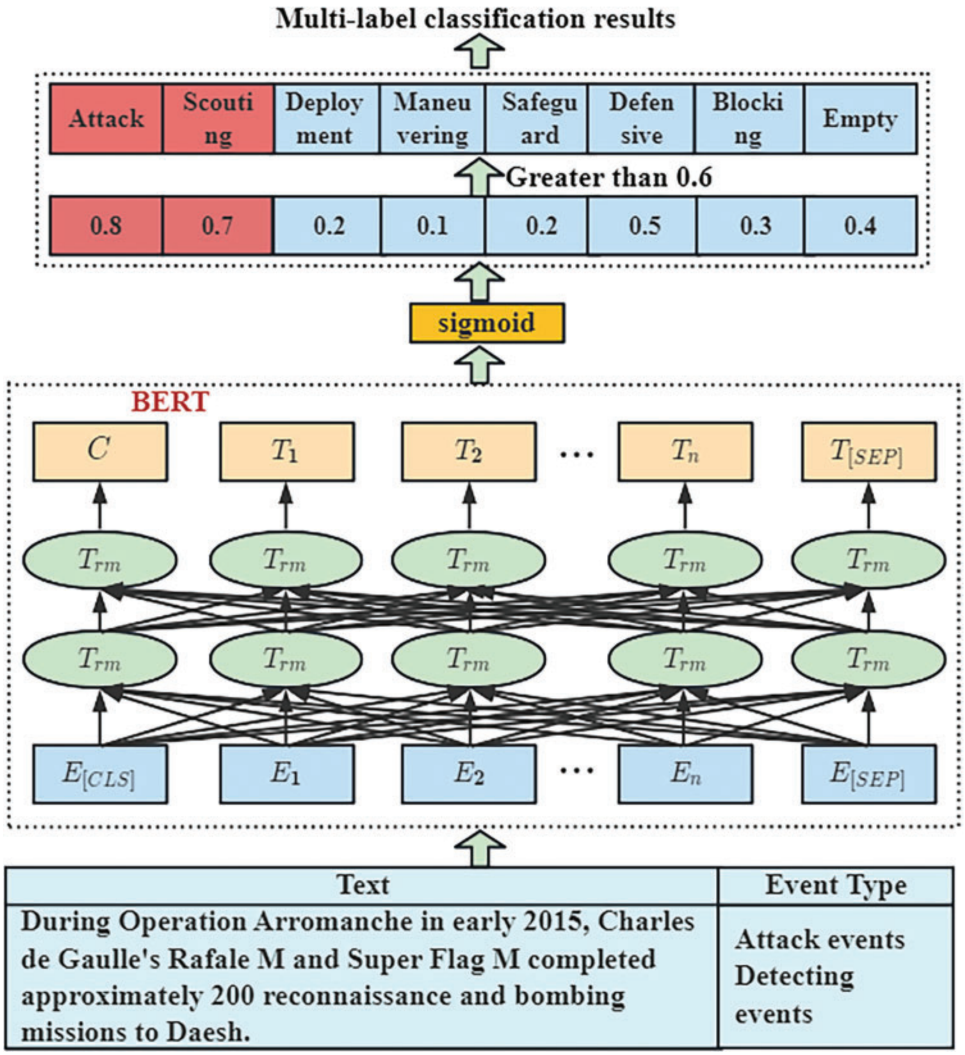


**Figure 3:** BERT model for multi-label text classification

The multi-label classification model is shown in Fig. 3. We encode the text using BERT to acquire a dynamic word vector representation of the sentence. Then, the encoded vectors are passed through a feedforward neural network that incorporates a sigmoid layer to classify the text and recognize the event type.

### *3.2 Event Argument Extraction*

Event argument extraction aims to extract the relevant arguments and the roles played by the arguments in an event. However, in the military domain, the scarcity of annotated data and the presence of overlapping event arguments pose significant challenges. To address these issues, we first use a full-domain random substitution data enhancement method based on arguments to perform event expansion while keeping the syntactic semantics unchanged. The main idea of the algorithm is to replace the arguments corresponding to the initiator, bearer, time, and location with arguments of the same type in the event text. For example, <Initiator: "Raider" Akinci latest UAV, Bearer: Russian "Armor-S1" Bomb and gun integrated air defense system, April 2021> replaced with <Initiator: French Mirage, Bearer: Syrian Palmyra, February 2020>. The trigger words, such as "attack," "strike," and "destroy," are replaced randomly. Then we extract the event arguments using the event argument extraction model. Fig. 4 shows an overview of the event argument extraction model, which is divided into four main phases: construction of input text, model pre-training, model building, and model fine-tuning.

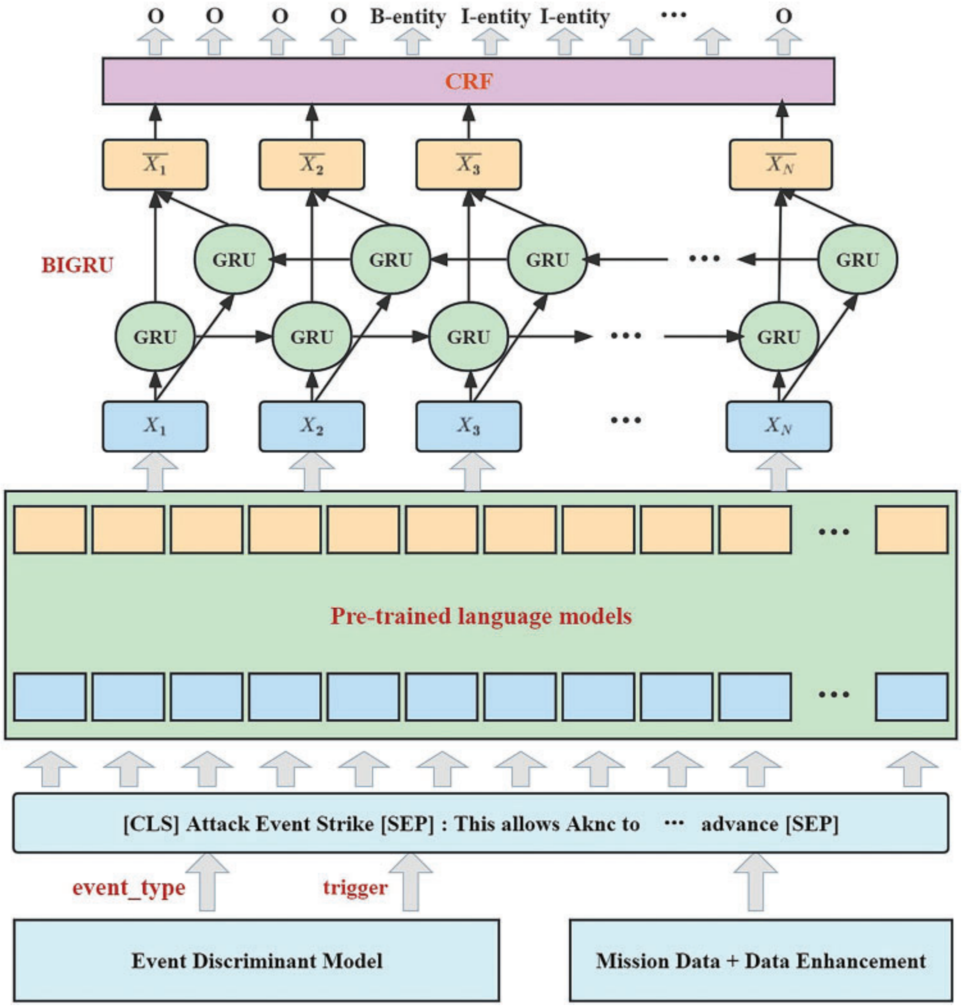


**Figure 4:** ERNIE-BIGRU-CRF model for event argument extraction

### *3.2.1 Construction of Input Text*

In the stage of constructing input text, we aim to address the issue of overlapping arguments between different event types within a sentence. Based on the event types and trigger words, we first use the event type recognition model output for integration, and then construct an a priori template of event types + trigger words as input text, implemented in the form of [CLS] + event types + trigger words + [SEP] + [text] + [SEP]. After using this approach to input text, fine-grained input samples with event trigger words can be obtained, enabling the model to more fully understand the semantic relationships between arguments. Meanwhile, trigger words that are not related to the event type are filtered using a multi-label classification model. The final result is generated by the trigger word extraction model and the multi-label classification model voting, which can further reduce the propagation error of the pipeline.

### *3.2.2 Model Pre-Training*

Sun et al. [29] stated that continued pre-training on an in-domain corpus can significantly improve the model's understanding of a specific domain. We divide the text into text sequences with a length of less than 300. To improve the adaptation and modeling abilities of the language model to the data, we continuously pre-trained the language model on the training texts.

### *3.2.3 Model Building*

The event argument extraction model is shown in Fig. 4. Firstly, ERNIE encodes the sentences to obtain a semantic feature vector of the sentences. Given the input token sequence $S = (s_0, s_1, \ldots, s_n)$, we incorporate each token into the transformer encoder to generate a word vector sequence $X = (x_{i1}, x_{i2}, \ldots, x_{in})$. This sequence is trained through the ERNIE model's embedding layer to obtain the word vector as follows:

$$\{W_{i1}, W_{i2}\ , \ldots, \ W_{in}\} = W_e\,(x_{i1}, x_{i2}, \ldots, x_{in}) \tag{1}$$

where $W_{in} \in R^{768}$ is the representation of the $n - th$ word and $W_e$ refers to the embedding layer's weight parameter.

Then, the vector is fed into BI-GRU to capture the long-range dependencies and output a sentence representation vector that incorporates deep semantic information as follows:

$$h_n^f = GRU\left(W_{in}, h_{n-1}^f\right) \tag{2}$$

$$h_n^b = GRU\left(W_{in}, h_{n-1}^b\right) \tag{3}$$

$$C = Concat\left(h_n^f, h_n^b\right) \tag{4}$$

where $h_n^f$ denotes the hidden state passed forward to the next node, $h_n^b$ denotes the hidden state passed backward to the next node, $h_{n-1}^f$ denotes the hidden state of the previous node forward, $h_{n-1}^b$ denotes the hidden state of the previous node backward, *Concat* represents the splicing of the forward and backward hidden layer state vectors, and $C$ is the output vector of the BiGRU layer.

Finally, the event arguments are labeled by the CRF layer and calculated as follows:

$$L = CRF\,(C) \tag{5}$$

$$Score = \sum\nolimits_{i=1}^{m} C_{i,y_i} + \sum\nolimits_{i=1}^{n-1} L_{y_i,y_{i+1}} \tag{6}$$

where $m$ is the number of label types, $C_{i,y_I}$ is the score of the tag $y_i$ of the $i - th$ token in the sequence, and $y_i$ represents the score of a transition from the tag $y_i$ to tag $y_{i+1}$.

The event argument extraction model calculates the loss value of the CRF layer on a sentence level, as follows:

$$Loss = -\log\left(\frac{P_r}{\sum_{i=1}^{m} P_m}\right) \tag{7}$$

where $P_m$ is the score corresponding to each predicted path, $m$ is the number of paths, and $P_r$ represents the score of ground truth.

### 3.2.4 Model Optimization

To enhance the performance of the event element extraction model, we incorporate a fine-tuning process that involves adjusting the learning rates. However, during experimentation, we observed that the model often converges within the desired range of 2e-5. To address this issue, we designed the learning rates using a layer-by-layer decreasing LayerRate [29], where lower learning rates are assigned to the lower layers of the network during the training phase. The learning rate is as follows:

$$\theta_{t}^{l} = \theta_{t-1}^{l} - \eta^{l} \cdot \nabla_{\theta^{l}} J(\theta) \tag{8}$$

$$\eta^{k-1} = \xi \cdot \eta^{k} \tag{9}$$

where $\eta^k$ represents the kth layer learning rate and $\xi$ represents the decay factor, $\xi = 0.95$.

In model training, the CRF layer failed to converge to the same learning rate. Due to the unequal coordination between the model and the CRF layer, the learning rate of the CRF layer has increased by 100 times. The Fast Gradient Method (FGM) [30] can form adversarial samples by adding perturbations to the embedding layer. Therefore, we use FGM to improve the robustness of the model to train a better-performing event argument extraction model.

## 3.3 Multi-Modal Knowledge Fusion

This article employs the BERT model to detect the types of arguments to achieve multi-modal knowledge fusion of text and images. Then, we use the YOLOv5 model to identify the object bounding boxes in the images and extract the corresponding type information. We propose a multi-modal label mapping method that jointly maps the classification results from argument identification and object detection to the same label space. By employing rule-based post-processing techniques, we establish links between spatial information (bounding box coordinates) and textual information (arguments) to effectively connect the visual and textual modalities.

### 3.3.1 Argument Recognition

In argument recognition, we first classify the arguments using a set of predefined rules. The unrecognizable arguments are then manually labeled using a crowdsourcing architecture, and the labeled dataset is transformed into a dataset for argument recognition.

Recently, pre-trained language models have achieved remarkable results on many natural language processing tasks. Given the input token sequence $W = (w_0, w_1, \ldots, w_n)$, we train this sequence through the embedding layer of the BERT model to obtain the feature vectors $T_i = (T_0, T_1, \ldots, T_n)$, as follows:

$$T_i = BERT\,(w_0, w_1, \ldots, w_n) \tag{10}$$

where $T_i$ represents the text vector representation obtained using the BERT pre-trained model, $i$ represents the $i - th$ text in the multi-modal dataset.

### *3.3.2 Object Detection and Recognition*

In object detection and recognition, image data annotation is combined with textual information. For example, "helicopter gunship" belongs to the category "aircraft". Therefore, each object in the image corresponds to a specific category. In this paper, we classify the weaponry in the image data into aircraft, ships, missiles, trucks, submarines, and six other "parent types".

To efficiently construct the object detection and recognition dataset, we use a combined human-machine label transformation method. The argument types are first converted to their "parent types" using an argument recognition model, then the data are labeled using Lambelme, and finally, the labeling errors are corrected through a manual review process.

Due to the overall better detection performance and faster processing speed of YOLOv5 in the object detection domain, we use YOLOv5 as the detection model. The basic model is fine-tuned by using military equipment object images to finally recognize the object bounding box of the image and the corresponding type. Before inputting the image into the model, we preprocess it by dynamically scaling it to a standardized size. Given input image dataset $I = (I_0, I_1, \ldots, I_n)$, the feature representation of each image output by the image feature extraction model is as follows:

$$V_i = YOLOv5\,(I_0, I_1, \ldots, I_n) \tag{11}$$

where $T_i$ represents the image features obtained through the YOLOv5 model, $i$ represents the $i - th$ image in the multi-modal dataset.

### *3.3.3 Multi-Modal Label Mapping*

In the multi-modal label mapping stage. Firstly, we fuse the output of the feature from independent variable recognition and object detection to obtain the fusion features of the text-image as follows:

$$A_i = T_i \oplus \; V_i \tag{12}$$

where $\oplus$ represents the fusion of feature vectors, $A_i$ represents the feature vector after the text and image fusion.

Then, we input the fused feature $A_i$ into the self-attention module and perform feature mapping through a fully connected layer, as follows:

$$M = \tanh\,(A_i) \tag{13}$$

$$\alpha = softmax\left(m^T M\right) \tag{14}$$

$$E_i = A_i\, \alpha^T \tag{15}$$

$$X_i = P\, E_i \tag{16}$$

where $m^T$ and $P$ are learnable parameters for the hidden layer, $\alpha$ is the standardized attention weight, $E_i$ is the feature representation output by the attention layer, and $X_i$ is the feature representation obtained through the fully connected layer.

Finally, the features are processed by the Softmax layer and calculated as follows:

$$F_i = softmax\,(X_i\; W + b) \tag{17}$$

where $F_i$ is the final classification result, $W$ is the weight matrix, and $b$ is the bias term.

In this stage, we classify the "initiator", "bearer" and "using the device" of the argument. Because "time" and "location" cannot be extracted from the image, these two arguments are assigned a value of “−1”. We use an object detection algorithm to identify the object bounding box and its type. A single image may contain multiple object boxes of the same type, thus we select the object box with the largest area. If the argument type corresponds to the image type, the object box coordinates are assigned to the argument. If they do not correspond, the value “−1” is assigned. But this method is not accurate enough. Therefore we use predefined rules to filter wrong arguments and overlapping arguments.

## 4 Experiments

In this section, we conduct a series of experiments to evaluate the effectiveness of our proposed approach. We first describe the implementation details, including data and hyperparameter settings. Then, we show the experimental results, including the performance of the model at each stage, and the entire multi-modal event element extraction approach.

### *4.1 Dataset and Evaluation Metrics*

In our experiments, we use the CCKS 2022 dataset oriented to the open-source multi-modal military event element extraction evaluation task. The dataset includes seven different event types: attack, scouting, safeguard, blocking, deployment, defensive, and maneuvering events. In this dataset, 1400 annotated military news texts are used as the training set, 200 annotated military news texts are used as the validation set, and 400 military news texts are used as the test set for evaluating the multi-modal event element extraction approach.

We use Precision (P), Recall (R), and F-Measure (F1) as the major metrics to evaluate the model performance of our models. A prediction is considered correct when it accurately identifies the event type, the event argument, and the location coordinates of the argument in the image. Regarding the correct coordinates of the argument in the image, one criterion is that the intersection ratio between the predicted position and the labeled position of the argument is greater than 0.5. If there is no corresponding coordinate for the argument in the image, the output is designated as −1 to indicate correctness. We use the event element matching F1 as the final evaluation metric with the following equation:

$$\mathrm{F1} = \frac{2 * P * R}{P + R} \tag{18}$$

where $P =$ the number of predicted correct event elements/number of all predicted event elements, and $R =$ the number of predicted correct event elements/number of all correctly annotated elements.

### *4.2 Implementation Details*

We use a PyTorch [31] and PaddlePaddle [32] based framework to implement the multi-modal event element extraction method. We divide 1400 training data into 7 copies using 7-fold cross-validation experiments. Suitable optimizers, learning rates, batch sizes, and weight recessions are used in our method. More detailed settings of the hyperparameters can be found in Table 2.

### *4.3 Experimental Results*

#### *4.3.1 Evaluations of Event Type Recognition Approaches*

We use a fusion of event multi-label classification and trigger word extraction models for event type identification. Tables 3 and 4 show the experimental results for the event multi-label classification task and the trigger word extraction task on the CCKS 2022 dataset, respectively. We can see from Table 3 that the BERT model improves the overall performance of the event multi-label classification task compared to other pre-trained language models. On the contrary, it is clear from Table 4 that the ERNIE model is better for the trigger word extraction task. This is because the trigger word extraction task requires more semantic information, and the ERNIR model can learn more semantic knowledge compared to other pre-trained models.

**Table 2:** Hyperparameters in the model

| Hyperparameters | Multi-tag classification | Event trigger word extraction | Event argument extraction | Argument recognition | Object detection |
|---|---|---|---|---|---|
| Model | BERT | ERNIE | ERNIE | BERT | YOLOv5 |
| Batch size | 8 | 16 | 16 | 32 | 16 |
| Epoch | 15 | 50 | 50 | 25 | 70 |
| Learning rate | 2e-5 | 2e-5 | 2e-5 | 3e-5 | 1e-5 |
| Dropout | 0.1 | 0.1 | 0.1 | 0.1 | N/A |
| Weight decay | 0.01 | 0.01 | 0.01 | 0.01 | 0.001 |
| Sequence length | 256 | 300 | 300 | 42 | 640 × 640 |
| Optimizer | Adam | Adam | Adam | Adam | SGD |

**Table 3:** Performance comparisons of multi-label classification algorithms

| Multi-label classification | Precision (%) | Recall (%) | F1 (%) |
|---|---|---|---|
| Xlnet-Base-Chinese [33] | 82.02 | 73.91 | 77.75 |
| Bert-Base-Chinese [13] | **85.07** | 74.31 | **79.32** |
| Ernie-3.0-Base-Zh [14] | 83.71 | 73.12 | 78.06 |
| RocBert-Base-Zh [34] | 81.17 | **76.68** | 78.86 |

**Table 4:** Performance comparisons of trigger words extraction algorithms

| Trigger words extraction | Precision (%) | Recall (%) | F1 (%) |
|---|---|---|---|
| Xlnet-Base-Chinese [33] | 83.69 | 70.36 | 76.56 |
| Bert-Base-Chinese [13] | 81.17 | 76.68 | 78.86 |
| Ernie-3.0-Base-Zh [14] | **84.12** | **77.47** | **80.66** |
| RocBert-Base-Zh [34] | 82.63 | 75.08 | 78.67 |

### *4.3.2 Evaluations of Event Argument Extraction Approaches*

Tables 5 and 6 show the experimental results of the event argument extraction task on the CCKS 2022 dataset, respectively. From Table 5, we can see that the ERNIE model slightly outperforms the other pre-trained models for the event argument extraction task. This is because ERNIE uses a knowledge masking strategy in the pre-training phase, which adopts three different granularity spans of token, phrase, and entity for masking in stages to learn semantic association information and entity boundary information. Therefore ERNIR has better performance in event argument extraction tasks.

**Table 5:** Performance comparisons of event argument extraction algorithms

| Event argument extraction | Precision (%) | Recall (%) | F1 (%) |
|---|---|---|---|
| Xlnet-Base-Chinese [33] | 52.33 | 61.53 | 56.56 |
| Bert-Base-Chinese [13] | 53.96 | 64.83 | 58.90 |
| Ernie-3.0-Base-Zh [14] | **54.60** | **66.44** | **59.91** |
| RocBert-Base-Zh [34] | 53.12 | 63.77 | 57.95 |

**Table 6:** Performance comparison of algorithms under different optimization strategies

| Event argument extraction | Model optimization strategy | F1 (%) |
|---|---|---|
| Ernie-3.0-Base-Zh [14] | Baseline | 59.91 |
| | +FGM | 60.33 |
| | +Add BIGRU and CRF Layer | 61.75 |
| | +Data enhancement | 62.17 |
| | +Trigger word splicing | 64.12 |
| | +Event type splicing | 64.79 |
| | +All | **65.74** |

Table 6 shows our improved performance using the model optimization method. We use FGM to do perturbation on embedding to further improve the model performance. The BIGRU + CRF model based on pre-trained ERNIE shows improvement in evaluation metrics compared to the pre-trained ERNIE model. This is because BIGRU can fuse deeper semantic information, and then compute probabilistic maximum label sequences by CRF, which solves the annotation bias problem. We use an argument-based full-domain random replacement data augmentation method to improve the category imbalance and improve the model performance. We constructed an a priori template based on event type + trigger word to solve the problem of overlapping parameters, thus effectively improving the F1 value of the evaluation metric. The trigger words we used are shown in Table 1. The magnitude of the loss value indicates the convergence of the model during the training process. Fig. 5 shows that the loss value of the event argument extraction model consistently remains at a low and stable level, indicating the model's excellent convergence performance on this dataset.

### *4.3.3 Evaluations of Multi-Modal Knowledge Fusion Approaches*

In the multi-modal knowledge fusion stage, two datasets were constructed for model evaluation based on the CCKS 2022 dataset. The first dataset includes the text of six "parent types" of argument entities. It comprises 1468 annotated argument entities as the training set, and 203 unannotated argument entities as the test set to evaluate the performance of the argument entity recognition approach. The second dataset consists of equipment images from the CCKS 2022 dataset. It includes

1400 images annotated with 2024 bounding boxes as the training set, and 200 images annotated with 318 bounding boxes as the test set for evaluating the performance of the object detection approach.

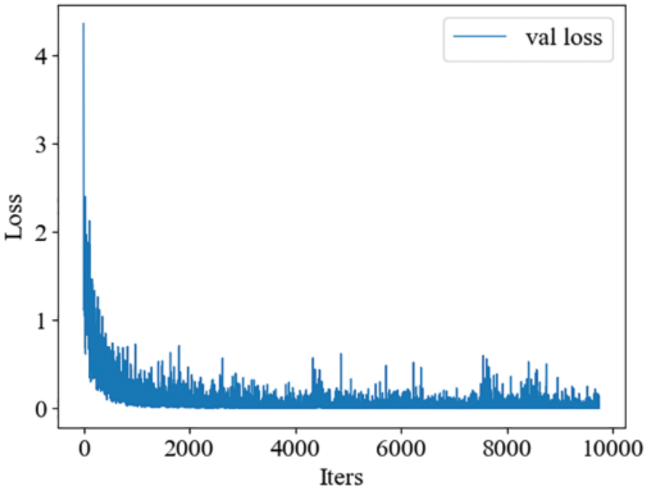


**Figure 5:** Change curve of loss value

(1) Argument Recognition. In this set of experiments, we used the first dataset containing the argument entities to train and evaluate the performance of the model using a 7-fold cross-validation approach. We use four pre-trained models, including BERT, XLNET, ERNIE, and RocBert, to identify six types of argument entities. The experimental results are shown in Table 7, where the BERT model obtained the best performance for P, R, and F1. This model accuracy can be used as a basis for label mapping in the multi-modal knowledge fusion phase.

**Table 7:** Performance comparisons of argument recognition algorithms

| Argument recognition | Precision (%) | Recall (%) | F1 (%) |
|---|---|---|---|
| Xlnet-Base-Chinese [33] | 93.42 | 93.70 | 93.56 |
| Bert-Base-Chinese [13] | **96.53** | 94.65 | **95.58** |
| Ernie-3.0-Base-Zh [14] | 95.51 | 93.43 | 94.45 |
| RocBert-Base-Zh [34] | 95.73 | **94.87** | 95.29 |

(2) Object Detection and Recognition. This experiment evaluates the object detection method on a second dataset consisting of object detection images. In this experiment, we used the YOLOv5 model for object detection and recognition. The model was fine-tuned using the equipment images, and the metric F1 value was evaluated, resulting in 0.753.

(3) Multi-modal Label Mapping. We evaluated the proposed multi-modal knowledge fusion approach using the CCKS 2022 dataset. This method fuses the text extracted from events and the images detected by the object. The final evaluation metric F1 value was 0.53403, obtaining competitive results on the CCKS 2022 dataset.

### *4.3.4 Performance Analysis of Model Usage Memory*

During the training process, we observed that the proposed model can have significant memory requirements, especially when working with larger datasets. These memory limitations can present challenges in real-world applications, particularly when deploying pre-trained models on devices with limited memory. To address potential memory issues in real-world scenarios, we explore several strategies. First, hardware accelerators like GPUs can be employed. For instance, we used a GeForce RTX 3090 in our implementation, which greatly improves memory utilization and accelerates model training. Additionally, techniques such as gradient checkpointing, gradient accumulation, and batch size reduction can be utilized to alleviate memory constraints. When optimizing memory usage, achieving the appropriate balance is crucial to ensure that the model's predictive ability is not compromised.

## 5 Conclusions

In this paper, we propose a multi-modal event extraction method based on knowledge fusion, to address the challenges of multi-modal event elements in the military domain. The method consists of three subtasks: event type recognition, event argument extraction, and multi-modal knowledge fusion. We first use a multi-label classification BERT model and a trigger word extraction ERNIE model to jointly recognize event types. Then the ERNIE-BIGRU-CRF model is used to extract event arguments. Finally, we use the BERT model to recognize argument entities and the YOLOv5 model to detect and recognize image objects for multi-modal knowledge fusion of images and text. In addition, we use a full-domain random substitution data enhancement method based on arguments to overcome the problem of insufficient labeled data in the military domain. We construct an a priori template of event types + trigger words to solve the argument overlap problem. The aforementioned methods demonstrate the ability to effectively extract event types and event elements from extensive multi-modal military data. This process enables the rapid extraction of valuable information, which holds great significance in enhancing the efficiency of military resource utilization and facilitating applied research on military knowledge.

The experimental results on the CCKS 2022 dataset demonstrate the effectiveness of the proposed method and yield competitive results. The extracted multi-modal event elements can be effectively used to support the informational analysis of military equipment. However, our proposed multi-modal knowledge fusion method suffers from propagation errors. Therefore, in the future, we will investigate fusing textual knowledge and image information in the feature space under small sample conditions to further improve the proposed multi-modal event element extraction method.

**Acknowledgement:** We are grateful to all those with whom we have enjoyed working on this and other related papers.

**Funding Statement:** This research was supported by the National Natural Science Foundation of China (Grant No. 81973695) and Discipline with Strong Characteristics of Liaocheng University–Intelligent Science and Technology (Grant No. 319462208).

**Author Contributions:** Study conception and design: Y. Xiang, X. Zhang; data collection: Y. Xiang; analysis and interpretation of results: Y. Xiang, Z. Zhang, Y. Jia; draft manuscript preparation: Y. Xiang, Y. Jia. All authors reviewed the results and approved the final version of the manuscript.

**Availability of Data and Materials:** The multi-modal military event extraction dataset used in this paper is available at https://github.com/xyy313/MMEE/tree/main/dataset.

**Conflicts of Interest:** The authors declare that they have no conflicts of interest to report regarding the present study.

## References

[1] J. Pan, K. Yi, H. Fang, Y. Zhao, K. Du *et al.,* "Analysis of military information construction under the background of artificial intelligence," in *E3S Web of Conf.*, EDP Sciences, Dali, China, vol. 235, pp. 2267–1242, 2021.

[2] Q. Guo, D. Zhu, Y. Wang and F. Wan, "Research and development of entity extraction based on information extraction," in *2021 Int. Conf. on Machine Learning and Intelligent Systems Engineering (MLISE)*, Chongqing, China, IEEE, pp. 366–369, 2021.

[3] L. Zhan and X. Jiang, "Survey on event extraction technology in information extraction research area," in *2019 IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conf. (ITNEC)*, Chengdu, China, IEEE, pp. 2121–2126, 2019.

[4] H. L. Trieu, T. T. Tran, K. N. Duong, A. Nguyen, M. Miwa *et al.,* "DeepEventMine: End-to-end neural nested event extraction from biomedical texts," *Bioinformatics*, vol. 36, no. 19, pp. 4910–4917, 2020.

[5] A. Ramponi, R. van der Goot, R. Lombardo and B. Plank, "Biomedical event extraction as sequence labeling," in *Proc. of the 2020 Conf. on Empirical Methods in Natural Language Processing (EMNLP)*, Punta Cana, Dominican Republic, Association for Computational Linguistics, pp. 5357–5367, 2020.

[6] C. Li, Y. Sheng, J. Ge and B. Luo, "Apply event extraction techniques to the judicial field," in *Adjunct Proc. of the 2019 ACM Int. Joint Conf. on Pervasive and Ubiquitous Computing and Proc. of the 2019 ACM Int. Symp. on Wearable Computers*, San Diego, California, USA, pp. 492–497, 2019.

[7] S. Mao, Z. Li, J. Cheng, W. Zhang, C. Wang *et al.,* "Research on information extraction in judicial field," in *2021 Int. Conf. on Artificial Intelligence and Electromechanical Automation (AIEA)*, Nanjing, China, IEEE, pp. 254–258, 2021.

[8] J. Deng, F. Qiao, H. Li, X. Zhang and H. Wang, "An overview of event extraction from Twitter," in *2015 Int. Conf. on Cyber-Enabled Distributed Computing and Knowledge Discovery*, Xi'an, China, IEEE, pp. 251–256, 2015.

[9] P. R. Rao and S. L. Devi, "EventXtract-IL: Event extraction from newswires and social media text in Indian languages," *FIRE (Working Notes)*, vol. 2266, pp. 282–290, 2018.

[10] C. Zhang, S. Soderland and D. S. Weld, "Exploiting parallel news streams for unsupervised event extraction," *Transactions of the Association for Computational Linguistics*, vol. 3, pp. 117–129, 2015.

[11] S. Han, X. Hao and H. Huang, "An event-extraction approach for business analysis from online Chinese news," *Electronic Commerce Research and Applications*, vol. 28, pp. 244–260, 2018.

[12] J. Liu, L. Min and X. Huang, "An overview of event extraction and its applications," *arXiv Preprint arXiv:*2111.03212, pp. 1–23, 2021.

[13] J. Devlin, M. W. Chang, K. Lee and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," *arXiv Preprint arXiv:*1810.04805, pp. 1–16, 2018.

[14] Y. Sun, S. Wang, S. Feng, S. Ding, C. Pang *et al.,* "ERNIE 3.0: Large-scale knowledge enhanced pre-training for language understanding and generation," *arXiv Preprint arXiv:*2107.02137, pp. 1–22, 2021.

[15] J. Chung, C. Gulcehre, K. Cho and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv Preprint arXiv:*1412.3555, pp. 1–9, 2014.

[16] J. Lafferty, A. McCallum and F. C. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *Int. Conf. on Machine Learning(ICML)*, Massachusetts, USA, pp. 282–289, 2001.

[17] X. Zhu, S. Lyu, X. Wang and Q. Zhao, "TPH-YOLOv5: Improved YOLOv5 based on transformer prediction head for object detection on drone-captured scenarios," in *Proc. of the IEEE/CVF Int. Conf. on Computer Vision*, Seoul, South Korea, pp. 2778–2788, 2021.

[18] E. Riloff, "Automatically constructing a dictionary for information extraction tasks," in *Proc. of AAAI*, Washington DC, USA: Citeseer, pp. 1–2, 1993.

[19] Y. Yang, Z. Wu, Y. Yang, S. Lian, F. Guo *et al.,* "A survey of information extraction based on deep learning," *Applied Sciences*, vol. 12, no. 19, pp. 91–96, 2022.

[20] Y. Chen, L. Xu, K. Liu, D. Zeng and J. Zhao, "Event extraction via dynamic multi-pooling convolutional neural networks," in *Proc. of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th Int. Joint Conf. on Natural Language Processing*, Denver, Colorado, USA, pp. 167–176, 2015.

[21] C. Tian, Y. Zhao and L. Ren, "A Chinese event relation extraction model based on BERT," in *2019 2nd Int. Conf. on Artificial Intelligence and Big Data (ICAIBD)*, Chengdu, China, IEEE, pp. 271–276, 2019.

[22] Q. Lyu, H. Zhang, E. Sulem and D. Roth, "Zero-shot event extraction via transfer learning: Challenges and insights," in *Proc. of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th Int. Joint Conf. on Natural Language Processing*, Cancun, Mexico, pp. 322–332, 2021.

[23] P. F. Felzenszwalb, R. B. Girshick, D. McAllester and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1627–1645, 2010.

[24] R. Girshick, J. Donahue, T. Darrell and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, Columbus, OH, USA, pp. 580–587, 2014.

[25] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, Las Vegas, NV, USA, pp. 779–788, 2016.

[26] L. Zhang, S. Fu, S. Jiang, R. Bao and Y. Zeng, "A fusion model of multi-data sources for user profiling in social media," in *Natural Language Processing and Chinese Computing: 7th CCF Int. Conf. (NLPCC 2018)*, Hohhot, China, pp. 3–15, 2018.

[27] Y. Ding, J. Yu, B. Liu, Y. Hu, M. Cui *et al.,* "MuKEA: Multimodal knowledge extraction and accumulation for knowledge-based visual question answering," in *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition*, New Orleans, LA, USA, pp. 5089–5098, 2022.

[28] K. Kowsari, K. Jafari Meimandi, M. Heidarysafa, S. Mendu, L. Barnes *et al.,* "Text classification algorithms: A survey," *Information*, vol. 10, no. 4, pp. 150–165, 2019.

[29] C. Sun, X. Qiu, Y. Xu and X. Huang, "How to fine-tune BERT for text classification?," in *China National Conf. on Chinese Computational Linguistics*, Kunming, China, Springer, pp. 194–206, 2019.

[30] T. Miyato, A. M. Dai and I. Goodfellow, "Adversarial training methods for semi-supervised text classification," *arXiv Preprint arXiv:*1605.07725, pp. 1–11, 2016.

[31] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury *et al.,* "PyTorch: An imperative style, high-performance deep learning library," *Advances in Neural Information Processing Systems*, vol. 32, pp. 1–12, 2019.

[32] Y. Ma, D. Yu, T. Wu and H. Wang, "PaddlePaddle: An open-source deep learning platform from industrial practice," *Frontiers of Data and Domputing*, vol. 1, no. 1, pp. 105–115, 2019.

[33] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. R. Salakhutdinov *et al.,* "XLNet: Generalized autoregressive pretraining for language understanding," in *Advances in Neural Information Processing Systems*, Vancouver, Canada: MIT Press, pp. 1–11, 2019.

[34] H. Su, W. Shi, X. Shen, Z. Xiao, T. Ji *et al.,* "RoCBert: Robust Chinese BERT with multimodal contrastive pretraining," in *Proc. of the 60th Annual Meeting of the Association for Computational Linguistics*, Dublin, Ireland, pp. 921–931, 2022.