IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received December 21, 2021, accepted February 3, 2022, date of publication February 10, 2022, date of current version February 17, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3150409

Research on Event Logic Knowledge Graph Construction Method of Robot Transmission System Fault Diagnosis

JIANFENG DENG[1], TAO WANG[1], ZHUOWEI WANG[2], JIALE ZHOU[1], AND LIANGLUN CHENG[2]

[1]School of Automation, Guangdong University of Technology, Guangzhou 510006, China
[2]School of Computer, Guangdong University of Technology, Guangzhou 510006, China

Corresponding author: Tao Wang (wangtao_cps@gdut.edu.cn)

This work was supported in part by the Key Program of NSFC-Guangdong Joint under Grant U1801263, Grant U2001201, and Grant U1701262; in part by the Natural Science Foundation of Guangdong Province under Grant 2020A1515010890 and Grant 2020B1515120010; in part by the Projects of Science and Technology Plan of Guangdong Province under Grant 2017B090901019 and Grant 2016B010127005; in part by the Science and Technology Plan of Foshan City under Grant 1920001001367; and in part by the Guangdong Provincial Key Laboratory of Cyber-Physical System under Grant 2020B1212060069.

ABSTRACT Knowledge graph technology has important guiding significance for efficient and orderly fault diagnosis of robot transmission system. Taking the historical robot maintenance logs of robot transmission system as the research object, a top-down fault diagnosis event logic knowledge graph construction method is proposed. Firstly, we define event arguments of fault phenomenon and fault cause events, define event argument classes and relation between classes, and construct an event logic knowledge ontology model. According to the event logic knowledge ontology, the fault diagnosis event argument entity and relation in the corpus are labeled, and an event logic knowledge extraction dataset is formed. Secondly, an event argument entity and relation joint extraction model is proposed. Using stacked bidirectional long short-term memory(BiLSTM) to obtain deep context features of text. As a supplement to stacked BiLSTM, self-attention mechanism extracts character dependency features from multiple subspaces, and uses conditional random field(CRF) to realize entity recognition. The character dependency features are mapped to the entity label weight embedding, and spliced with deep context features to extract relations. Bidirectional graph convolutional network(BiGCN) is introduced for relation inference, graph convolution features are used to update deep context features to perform joint extraction in the second phase. Experimental results show that this method can improve the effect of event argument entity and relation joint extraction and is better than other methods. Finally, an event logic knowledge graph of robot transmission system fault diagnosis is constructed, which provides decision support for autonomous fault diagnosis of robot transmission system.

INDEX TERMS Event logic knowledge graph, fault diagnosis ontology, event argument knowledge extraction, stacked BiLSTM, self-attention, BiGCN.

I. INTRODUCTION

With the rapid development of intelligent manufacturing, industrial robots play an increasingly important role in the production process of enterprises, which can perform production tasks more efficiently and accurately. Generally speaking, the structure of industrial robot is complex, and its system equipment fault problem is very prominent [1]. The transmission system is an important part of robot, and its main function is to provide power for robot [2]. If the transmission system have faults, the working efficiency of robot will be reduced, and even lead to casualties. Therefore, intelligent fault diagnosis of transmission system is an important means to ensure the safe and stable operation of robot. At present, in the knowledge economy environment, knowledge-based fault diagnosis is a new type of intelligent fault diagnosis method [3]. Enterprises pay more attention to the acquisition and utilization of knowledge, obtain fault diagnosis knowledge from the fault diagnosis data accumulated by enterprises and expert diagnosis experience, then organize these knowledge into a visual representation can help the diagnosis personnel fully grasp the key information of fault diagnosis,

The associate editor coordinating the review of this manuscript and approving it for publication was Mu-Yen Chen.

and improve the efficiency of fault diagnosis. It can ensure the safe and stable operation of robot equipment and improve enterprise production efficiency.

At present, the research work of robot fault diagnosis based on knowledge is limited. Hsu *et al.* [4] used nelson rule to detect robot faults online, and perform diagnostic operations when the robot equipment is abnormal. Zhang *et al.* [5] designed a expert system for welding robots fault diagnosis, using fault tree analysis to search for fault knowledge and reason about the importance of fault events. Castellano-Quero *et al.* [6] proposed a reasoning system that integrates expert knowledge, external information and other knowledge sources, then has good detection performance for robot abnormalities. Generally speaking, most of the above-mentioned research use rules and expert knowledge reasoning for fault diagnosis, and the knowledge content is not highly structured. With the accumulation of knowledge, there are problems such as rule design conflicts, event knowledge have high redundancy, weak self-reasoning and updating learning ability [7]. Therefore, it is necessary to design fine-grained knowledge representation method, form knowledge structured network and optimize fault intelligent diagnosis.

As a method of knowledge representation, knowledge graph represents structured knowledge with triples [8]–[10] to form a network knowledge structure. It can uniformly represent and integrate the knowledge with low degree of structure. In addition, knowledge graph also has self-reasoning and updating learning ability [11], which can mine other hidden knowledge from the existing knowledge. It can realize the functions of efficient knowledge query and reasoning. Knowledge graph includes a pattern layer and a data layer, and its construction methods are usually divided into three modes: top-down, bottom-up, and mixed [12]. The pattern layer is mainly used for ontology conceptual construction of domain knowledge. The data layer mainly obtains knowledge instances through knowledge extraction methods. In the field of robot transmission system fault diagnosis, due to the higher accuracy of ontology and fixed format of corpus, it will be more effective to construct knowledge graph from top-down. However, there are three aspects to be improved in the construction of robot transmission system fault diagnosis knowledge graph:

- In the field of transmission system fault diagnosis, in addition to qualitative knowledge, for example, the motor is consist of bearings. It also contains a lot of event logic knowledge composed of event trigger words or event causality, for example, the aging of insulation leads to electrification of shell. There are temporal and event logic relations in this process. However, the ability of existing knowledge graph construction to reflect the logical knowledge is weak [13]. Therefore, it is necessary to construct an event logic knowledge graph that combines qualitative and event to better realize event logic inference and reduce manual intervention in fault diagnosis.
- The robot transmission system fault diagnosis corpus mainly describes the fault phenomenon and corresponding causes. It is event records of equipment fault in a specific time. The existing fault diagnosis event ontology construction methods define fault phenomenon events or fault cause events as event classes, and events are usually described in unstructured corpus, it easily leads to a coarser granularity of extracted knowledge instances, which is not conducive to subsequent knowledge inference. Therefore, it is necessary to further decompose event class and construct a fine-grained fault diagnosis event logic ontology.
- In terms of knowledge extraction methods, compared with public datasets, because enterprises do not allow robots to run in fault state for a long time, the recorded fault diagnosis corpus is less [14], the sentence length is limited, and the problem of sparse semantics is faced [15]. Therefore, it is necessary to design a multi-level structure network to obtain multi-level sequence semantic abstract features from multiple subspaces in order to better understand sentence semantic structure. In addition, the existing entity and relation joint extraction methods mainly use entity recognition results to assist relation extraction, but do not make full use of the interconnection between entity recognition and relation extraction. Therefore, it is necessary to design a relation inference method to introduce the result information of relation extraction into entity recognition, enhance the interconnection between subtasks, and improve the performance of joint extraction.

Based on the above motivations, this paper proposes a construction method of transmission system fault diagnosis event logic knowledge graph, collects fault diagnosis event description corpus of robot transmission system, and uses a top-down method to construct event logic knowledge graph. Firstly, a fine-grained fault diagnosis event logic ontology model is established, and the corpus is labeled according to the ontology. Secondly, a self-attention-based stacked BiLSTM with label weight embedding and graph convolution network(SBALGN) is proposed for event argument entity and relation joint extraction. Finally, the extracted knowledge are used to construct fault diagnosis event logic knowledge graph. The contributions of this paper are as follows:

- We collect a fault diagnosis event description corpus of robot transmission system, and a fine-grained event logic knowledge ontology is constructed. The fault diagnosis event argument entities and relations are labeled. An fault diagnosis event argument entity and relation joint extraction dataset is established.
- We propose a fault diagnosis event argument entity and relation joint extraction model, which is divided into two phases. Firstly, by stacked BiLSTM and self-attention mechanism, we obtain deep context features and character dependency features from text itself to enhance the ability of understanding sentence semantic structure. Then CRF and the introduction of entity label

weight embedding splicing with deep context features are used for event argument entity recognition(EAER), event argument relation extraction(EARE), respectively. Finally, BiGCN relation inference uses relation prediction results to aggregate entity nodes and update deep context features for joint extraction in the second phase, This model can add relation information to entity recognition, so as to improve the performance of joint extraction.

- We comprehensively evaluate the performance of SBALGN. The implementation results show that this method is better than the latest entity and relation joint extraction methods in recent years. Through the above-proposed SBALGN, the event logic knowledge are extracted, and the event logic knowledge graph of robot transmission system fault diagnosis is initially constructed.

The remainder of the paper is organized as follows. Section 2 mainly introduces the knowledge applied by the proposed event logic knowledge graph construction method and gives a literature review. Section 3 introduces the construction method of event logic knowledge graph in detail. Section 4 presents the experimental results and analysis of SBALGN, and realizes the visualization of event logic knowledge graph. Finally, Section 5 gives the experimental conclusions and future work to be done.

II. RELATED WORK

In this section, we will introduce related work on the construction of knowledge graph, ontology construction of fault diagnosis and knowledge extraction methods, which are the research foundation of this paper.

A. KNOWLEDGE GRAPH CONSTRUCTION

Knowledge graph mainly describes the concepts and relations of the physical world. It is first applied to the general field. Many large-scale general domain knowledge graphs have appeared on the Internet, such as DBpedia [16], Yago [17], wikidata [18] etc. General knowledge graph contains a lot of facts and common sense encyclopedia data, which mainly reflects the nominal entities and their deterministic relations, emphasizes the breadth of knowledge.

Domain-specific knowledge graph is different from general knowledge graph. Abu-Salih [19] pointed out that the domain knowledge graph is an explicit conceptualization of high-level subject domain and its specific subdomains, which is expressed by semantically related entities and relations. In the process of subject domain conceptualization, expert knowledge is needed to help construct specific domain ontology and determine the subject domain. On this basis, the semantic network is constructed by extracting pre-defined specific entities and relations. The domain-specific knowledge graph emphasizes the depth of knowledge, and the knowledge contained is highly targeted and professional. Representative specific fields include medical [20], economics [21], education [22], etc.

The above-mentioned domain-specific knowledge graph mainly focuses on entities and attribute relations defined in the specific domain to solve where, what and other problems. In the field of robot transmission system fault diagnosis, there are a large number of event description knowledge and event logic knowledge. The knowledge graph of entity and attribute relation type is not ideal for the expression of this part knowledge, and it cannot solve problems such as why. Event knowledge graph takes events as the core concept, pays attention to predicate triggered events and their logical relations. Not only reflects the essence of events, but also shows the development law of events, and focuses on solving why and other problems. As a new type of knowledge graph, it has already attracted the attention of researchers. Hoang Long and Jung [23] proposed a knowledge graph of social events, taking event as concept center, and taking people, time and place in events as event attributes, which can provide an understanding and traceability of social events. Guo *et al.* [24] proposed a construction method of financial event knowledge graph to enhance event extraction. The experimental results show that the construction of knowledge graph can effectively improve the performance of event extraction. Ringsquandl *et al.* [25] proposed a construction method of knowledge graph of equipment operation and maintenance events based on machine learning, in order to predict the missing knowledge in knowledge graph. Rospocher *et al.* [26] proposed an event-centric knowledge graph, established logic relation between events, and reconstructed the development and evolution of events. Li *et al.* [27] proposed a concept of event evolutionary graph, which described logic relation between events, and used it to discover the evolution law of events and predict subsequent events.

In summary, event knowledge graph has been used in manufacturing, financial industry and other fields. It mainly takes event as nodes and edges as relations between events, reflecting the logic relation between events, but it has not been applied in the field of robot transmission system fault diagnosis. The research work of this paper is different from the above. This paper proposes a knowledge graph construction method combining qualitative knowledge and event knowledge, which provides a knowledge base of transmission system fault diagnosis, so as to improve the efficiency and accuracy of fault diagnosis.

B. ONTOLOGY CONSTRUCTION OF FAULT DIAGNOSIS

The pattern layer of event knowledge graph is mainly constructed by the methods of ontology modeling. Ontology is to give a clear definition of knowledge unit in domain knowledge. With the continuous development of knowledge ontology, different ontology description languages have been produced. At present, OWL2 has become the recommended description language for ontology modeling [28]. In addition, part of the research work focuses on the ontology construction methods, including skeleton method [29], methodology method [30], seven-step method [31] and so on, among which seven-step method is more mature and effective than other ontology modeling methods [32], [33], so seven-step method is one of the most popular modeling methods, which can combine with protg tools to achieve ontology modeling. Therefore, this paper refers to the seven-step method to construct fault diagnosis ontology of transmission system.

In recent years, in order to achieve knowledge extraction and sharing, the research and application of fault diagnosis event ontology are increasing. Zhou *et al.* [34] proposed an ontology model of rolling bearing fault diagnosis, taking fault reasons, fault phenomenon, auxiliary measures, object attributes as the main concepts in the field of fault diagnosis, and defining logical relations such as cause. Taking the fault diagnosis of production line equipment as an example, Geng and Fu [35] constructed an ontology model for fault phenomenon, fault reasons, fault sources and maintenance schemes, so as to prepare for subsequent troubleshooting. Wang *et al.* [36] took fault diagnosis of fuel injection pump equipment as an example, defined domain ontology with equipment, functions, attributes, attribute values, function flows and fault symptoms, defined the relations between concepts as causality or dependency. Zhou *et al.* [37] proposed an ontology-based machine tool fault diagnosis platform, including fault types, detection and identification methods, fault reasons, and maintenance strategy knowledge. It is used to realize diagnosis and improve manufacturing strategy. Zhou *et al.* [38] proposed a knowledge modeling method of machine tool fault diagnosis, summarized four key concepts in the field of fault diagnosis, including fault phenomenon, fault maintenance, fault reason and fault location, defined concepts and conceptual attributes as ontology classes, so as to establish the core ontology of machine tool fault diagnosis. Xu *et al.* [39] proposed a fault diagnosis ontology model for loaders and defined five fault diagnosis concepts, including fault modes, faulty equipment, fault maintenance, parameters and fault phenomenon, which provided a general method for loader fault diagnosis. Tsalapati *et al.* [40] proposed new fuel cell system monitoring (FCSM) ontology, which is built around fuel cell structure information, component attributes, and historical fault diagnosis rules. Nunez and Borsato [41] proposed predictive health management (PHM) ontology model, which can be used for various types of machinery and store the knowledge contained in equipment activity events, including device fault mode, potential fault causes, and fault location, etc., it can intervene in equipment maintenance in time. Liu *et al.* [42] proposed a semantic web-based machine tool fault diagnosis knowledge expansion method, based on the OKM-MTFD core ontology to construct different types of machine tool fault diagnosis models, and realized efficient collection of the acquired machine tool fault diagnosis knowledge.

In summary, in recent years, there have been a certain amount of research results on the construction of knowledge ontology for equipment fault diagnosis, mainly focusing on the event concept of equipment fault structure, phenomenon and cause, then taking the specific equipment structure and phenomenon as instances for knowledge query and reasoning. However, equipment phenomenon and equipment causes are mainly described in unstructured event mode. If they are regarded as event entity knowledge, the degree of knowledge structure is still not ideal. The ontology proposed in this paper is different from the above-mentioned research methods. Decomposes the concept of fault phenomenon and fault cause to form concept classes of equipment and fault state. Some fault states are reflected by equipment attributes and attribute values. Therefore, the fault states concept class can be decomposed into attributes and attribute values. Through the connection of event trigger word relation, event logical relation, the construction of fault diagnosis knowledge ontology is realized.

C. KNOWLEDGE EXTRACTION METHODS

The construction of knowledge graph data layer mainly uses knowledge extraction methods to obtain knowledge instances from unstructured corpus. For the construction of event logic knowledge graph, event logic knowledge extraction is mainly to extract event argument entities, event argument entity relations. The existing knowledge extraction methods mainly focus on qualitative entity and relation extraction, which is divided into pipeline method and joint extraction method. The pipeline method usually ignores the interconnection between entity recognition and relation extraction tasks. The joint extraction method combines entity recognition and relation extraction together, which can interact information with each other to improve the performance. Therefore, Joint extraction methods have attracted the attention of researchers. Zheng *et al.* [43] proposed a new annotation strategy, which transformed entity relation joint extraction into labeling problem, and proposed BiLSTM to extract entity relations, which achieved good results on the NYT dataset. However, this method still has shortcoming on the extraction of overlapping relations. Katiyar and Cardie [44] proposed a recurrent neural network(RNN) based on attention mechanism, a multilayer BiLSTM to obtain context feature recognition entities, and use the attention mechanism to output relation labels. Proved superior to feature-based models on the ACE corpus. Giannis *et al.* [45] propose general joint extraction model, which uses BiLSTM to extract the context features of text, CRF is used for entity recognition, and the relation extraction task is transformed into multi-head selection problem, which can extract multiple relations involved in an entity. Experimental results on CoNLL04 dataset show that the model is better than the previous models. Fu and Ma [46] propose GraphRel for entity and entity relation joint extraction. Through relation weighted GCN, it can improve the prediction of overlapping relations. The performance of GraphRel on the NYT dataset is 3.2% higher than previous work. However, this method uses sentence dependency as adjacency matrix of GCN, which may not be suitable for all languages. Zhou *et al.* [47] propose joint extraction model based on attention mechanism. The entity embedding vector extracted from the pre-trained entity recognition model is used as entity features for relation classification. In the relation classification, attention mechanism is introduced to select important information for prediction. The validity of the model is confirmed on NYT dataset. Zhang *et al.* [48] design BiLSTM to extract context features for entity recognition, and further passed context features and entity label features to CNN for relation extraction. The validity of the model is proved on CoNLL04 dataset. Cao *et al.* [49] combine the joint extraction of drug entities and relations into sequence labeling problem, and propose a new labeling strategy. BiLSTM-CRF is used for labeling. Drug entities and relations are identified according to the labeling results. This method can alleviate the problem that overlapping entity relations cannot be extracted.

From the above research work, it can be seen that most entity and relation joint extraction models are mainly applied to English public datasets. The entity category is predicted by CRF or LSTM, then converted into label embedding to be added to the relation extraction task. The research work in this paper is different from the above work. Inspired by [46], [50], [51], stacked BiLSTM is used to improve the ability of shallow BiLSTM to capture context information. In EAER subtask, in addition to the necessary text context features, it is also necessary to obtain the dependencies between characters from multiple perspectives to better understand the sentence structure. Therefore, this paper introduces a self-attention mechanism to analyze text content, which has an improved effect on EAER. For the task of EARE, we propose entity label weight embedding and deep context features splicing for relation prediction. After the first phase of extraction, the BiGCN relation inference is proposed to add the result information of relation prediction into EAER in the second phase, which can not only improve the performance of EAER, but also improve the performance of EARE. Finally, the performance of joint extraction is improved.

III. METHODOLOGY

In this section, we first introduce a fault diagnosis event logic ontology construction of robot transmission system, then introduce a fault diagnosis event argument entity and relation labeling strategy, and finally introduce SBALGN model proposed in this paper.

A. FAULT DIAGNOSIS EVENT LOGIC ONTOLOGY CONSTRUCTION

The core of ontology defines knowledge concepts and the relation between concepts. Traditional event ontology adds the description of event concepts and their relations on the basis of ontology. At present, the seven-step method of ontology construction is widely used to construct ontology. The seven-step method mainly includes determining the domain and scope, considering the reuse of existing ontologies, listing important terms in the field, defining classes and their hierarchy, defining class attributes and relations, defining attribute facets, and creating instances. In the seven-step method, the core steps include listing important terms in the field, defining classes and their hierarchy, defining class attributes and relations, and defining attribute facets. In this paper, the domain knowledge of fault diagnosis is represented by event logic. For the construction of fault diagnosis ontology, it has been determined that the domain field of ontology is robot transmission system fault diagnosis. The fault diagnosis corpus expresses fault cause events and fault phenomena events, mainly involving text format data, and there is no suitable ontology that can be directly reused. Refer to the core steps of the seven-step method, combined with the knowledge expression needs of event logic, because the fault diagnosis case corpus does not involve specific objects, the ontology construction proposed in this paper does not include defining class attributes and attribute facets, and is divided into two steps. The first is the definition of fault diagnosis event argument, which mainly defines the argument element terms object(O), trigger word(T), status(S) that constitute an event. The second is to define the class concept and class relation of event arguments. The class concept definition mainly describes the definition of argument class and the ontology class concept construction of fault diagnosis events. The class relation definition mainly describes the definition of event trigger word relations and logic relation between events, the construction of ontology relation concepts. It will be described in detail below.

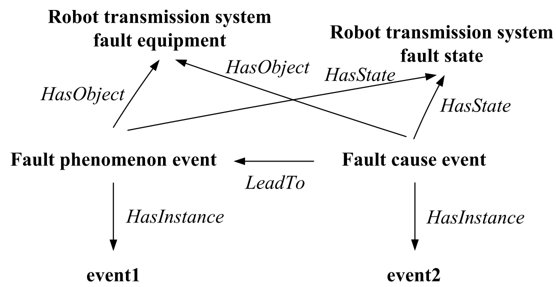


FIGURE 1. Fault diagnosis event and argument modeling.

1) FAULT DIAGNOSIS EVENT ARGUMENT DEFINITION

As mentioned above, the fault phenomenon and corresponding fault cause are fault events of equipment. Formally, e is used to represent an event. The event is composed of three elements, and tuples can be used to represent formula 1.

$$e = < O, T, S > \tag{1}$$

Among them, O represents fault object, corresponding to equipment words in text, such as motors, bearings and so on. T represents event trigger element, which corresponds to the event trigger word in text, such as appear and so on. S is state expression of equipment, which mainly includes fault state words. There are logic relations between events, such as lead to and so on. The conceptual knowledge modeling of fault diagnosis events and event argument is shown in figure 1. As can be seen from figure 1, the concept of fault diagnosis event includes fault phenomenon event concept and fault cause event concept. There is a *LeadTo* relation between them. Event1 and event2 are event instances, each event includes corresponding fault equipment objects and equipment fault status. They are the arguments of fault diagnosis event.

2) FAULT DIAGNOSIS EVENT ARGUMENT CLASS AND RELATION DEFINITION

Robot transmission system fault diagnosis corpus is mainly composed of historical fault diagnosis events. Robot transmission system can be divided into four levels from top to bottom, including equipment, sub-equipment, components and parts. In historical fault description events, part of fault description events are both phenomenon events and cause events. For example, damp winding leads to electrification of shell. Damp winding is not only a phenomenon, but also a cause of electrification of shell. In order to solve the problem of phenomenon and cause concepts coarse granularity in current conceptual modeling methods, this paper decomposes phenomenon and cause concepts into equipment structure class and fault state class. In addition, the causes and phenomenon of equipment faults can be expressed by equipment attributes and fault attribute values. Therefore, the fault diagnosis ontology model defines three classes, including fault equipment structure, equipment attributes and equipment state values, which belong to the fault diagnosis event argument class. Among them, the equipment fault attribute includes fault attribute value subclass, the fault equipment structure includes sub equipment subclass, component subclass and part subclass. Each layer of equipment structure has corresponding equipment fault attributes.

When concept classes are defined, there are corresponding class relation between each conceptual classes. This paper defines event argument class relations, including *consist_of*, *lead_to*, *has_attribute*, and *appear*. Among them, *consist_of* represents relation between equipment and sub-equipment, *has_attribute* represents relation between equipment structure and attribute, these two kinds of relations belong to qualitative knowledge relations. *appear* represents event trigger word relation between equipment structure and state, this kinds of relation belong to event trigger word relation, *lead_to* represents event logic relation between states, this kinds of relations belong to event logic relation. Because not all state classes have *lead_to* relation, so this paper uses restriction in protg software to limit *lead_to* relation between states. In addition, some fault attribute values which lead to the appearance of equipment fault state, so fault attribute values and equipment fault states also have *lead_to* relation, and restriction operation is also used to restrict.

After defining classes and class relations, the fault diagnosis event logic ontology model is constructed by protg tools, and knowledge are obtained from unstructured texts through knowledge extraction methods as knowledge instances of ontology model. Figure 2 show the structure of fault diagnosis event logic ontology model. In figure 2, motor and overload are creation instances of equipment and state classes respectively, and there is a *appear* event trigger word relation between them.

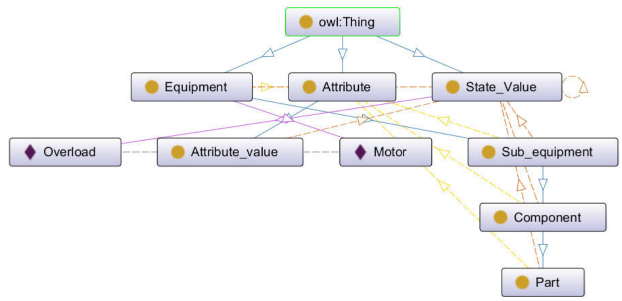


FIGURE 2. Structure of fault diagnosis event logic ontology model.

B. FAULT DIAGNOSIS EVENT ARGUMENT ENTITY AND RELATION LABELING STRATEGY

Inspired by [52]–[54], fault diagnosis event argument entity and relation joint extraction is transformed into sequence labeling task. This task draws on the labeling method of English open datasets, which consists of five parts, which are entity boundary, entity category, relation category, character position and tail entity position. In which the entity boundary uses BIO mode, where *B* represents the beginning of entity, *I* represents the middle and the end of entity, *O* represents non-entity. Event argument entities and relations are predefined by experts with background knowledge. Event argument entities are mainly divided into fault object and its composition, fault state, fault attribute and its fault attribute value. Event argument entity relation is mainly divided into qualitative knowledge relation, event trigger word relation and event logic relation.

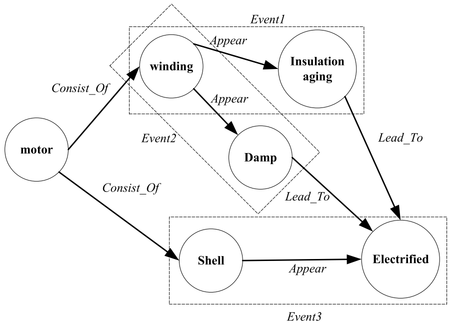
According to the event logic ontology model proposed in section 3.1, this paper labels 7 entity categories(*Equipment*, *Sub_equipment*, *Component*, *Part*, *Attribute*, *Attribute_value* and *Status_value*) and 4 relation categories(*Consist_Of*, *Lead_To*, *Appear* and *Has_Attributes*), respectively. Among them, *Equipment*, *Sub_equipment*, *Component*, *Part* correspond to the above-mentioned fault object and its composition, *Attribute*, *Attribute_value* correspond to the above-mentioned fault attribute and its fault attribute value, and *Status_value* corresponds to the above-mentioned fault state. In the relation category, *Consist_Of*, *Has_Attributes* correspond to the above-mentioned qualitative knowledge relation, *Appear* corresponds to the above-mentioned event trigger word relation, and *Lead_To* corresponds to the above-mentioned event logic relation. The relation category is marked in the head entity. The character position represents the position of character in sentence. The tail entity position represents the position of the tail character of the tail entity corresponding to the relation category.

Figure 3(a) and figure 3(b) take a sentence sample as an example, respectively explain our labeling strategy in this paper and an English translation example of extracting event argument entity and relation triples from sentence sample according to the labeling strategy. Given a sentence: the motor winding is damp and the insulation aging leads to

Text	电	机	绕	组	受	潮	，	绝
Character position	0	1	2	3	4	5	6	7
Event argument entity label	B-Equipment	I-Equipment	B-Component	I-Component	B-Status_value	I-Status_value	O	B-Status_value
Event argument relation label	N	Consist_Of,Consist_Of	N	Appear,Appear	N	Lead_To	N	N
Tail entity position	0	3,13	2	5,10	4	15	6	7

Text	缘	老	化	使	外	壳	带	电
Character position	8	9	10	11	12	13	14	15
Event argument entity label	I-Status_value	I-Status_value	I-Status_value	O	B-Sub-equipment	I-Sub-equipment	B-Status_value	I-Status_value
Event argument relation label	N	N	Lead_To	N	N	Appear	N	N
Tail entity position	8	9	15	11	12	15	14	15

(a)



(b)

FIGURE 3. Sample labeling and event argument entity and relation extraction example.

electrification of shell. The sentence contains seven triples and three event instances, *Consist_Of*, *Appear*, and *Lead_To* are the pre-defined relation categories. Each character is labeled according to entity and relation information. Since some entities are composed of multiple characters, and triple includes head entity and tail entity, the relation category is marked on the tail character of the head entity. If the entity does not have any relations, the relation category is marked as N. When there is entity relation between entities, mark the tail character position of the tail entity in sentence. For the case where the same entity involves multiple triples, mark all relations on the tail character of the head entity, and mark the position of the relation corresponding to the tail character of the tail entity.

In the definition of ontology, fault diagnosis is mainly carried out through fault causes and fault representation. Fault cause events and fault representation events are mainly composed of equipment, trigger words, fault status or fault attribute values, including three types of event elements. Equipment and fault status are connected by trigger word, equipment and fault attribute are connected by *Has_Attributes*, and the relation between events is connected by *Lead_To*. In the case corpus collected in this paper, each fault has its corresponding fault cause and fault representation. In addition, we can find that different fault causes can cause the same or similar fault phenomenon, and we can also find that a fault cause can cause multiple different fault phenomenon, but these all represent different fault modes. In summary, when the faults are different, the corresponding fault causes and fault representations are not exactly the consistent. Therefore, in this corpus, it is not found that there are different faults corresponding to fault causes and fault representations that are exactly the consistent.

During data labeling, we arrange 2 labeling staff to manually label the case corpus. Before labeling, they need to agree on the agreement of labeling, according to the previously developed labeling strategy, when the corpus labeling is completed, they need to exchange the labeling corpus for comparison and inspection. If they have any objection to the labeling results, they will conduct a centralized discussion and make manual revisions. Finally, there will be an additional annotator to participate in the inspection and final discussion, until all the annotators agree and complete the annotation.

C. EVENT ARGUMENT ENTITY AND RELATION JOINT EXTRACTION MODEL

SBALGN includes character embedding layer, stacked BiLSTM layer, event argument entity and relation extraction layer, and BiGCN entity relation inference layer. Firstly, the character feature representation is obtained through character embedding layer, which is used as the input of stacked BiLSTM layer. Secondly, stacked BiLSTM layer is used to obtain deep context features, and self-attention mechanism is used as a supplement of stacked BiLSTM to obtain character dependency features. CRF is used to recognize event argument entities, and the character dependency features are mapped to label weight embedding, which splice with deep context features for EARE. Finally, BiGCN entity relation

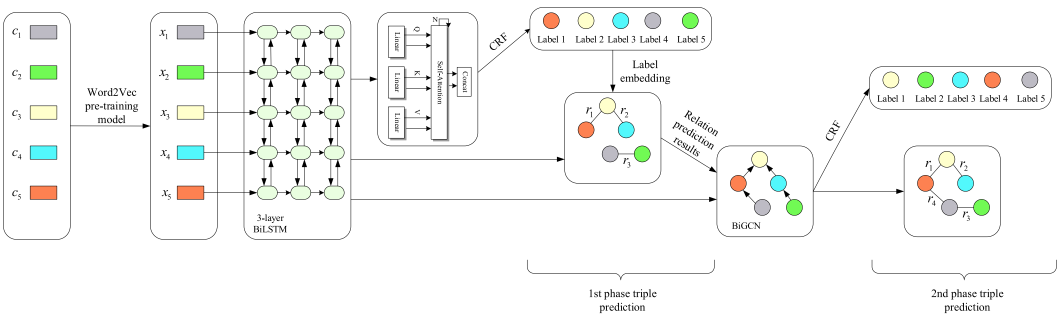


FIGURE 4. The overall structure of our proposed SBALGN model.

inference layer infers the relation prediction results, updates deep context features, and performs the second phase of event argument entity and relation joint extraction. The overall structure of SBALGN is shown in figure 4.

1) CHARACTER EMBEDDING LAYER

For Chinese sequences, the results of Chinese word segmentation may be different from the actual results, causing entity recognition errors to propagate in the model. Therefore, this paper uses character feature vector sequence to represent sentence sequence. The character feature vector training model adopts word2vec pre-training language model [55]. Word2vec has two training modes, including skip-gram and CBOW [56]. When the training corpus is small, the skip-gram mode works better, and when the training corpus is large, the CBOW mode is used [57]. Therefore, this paper adopts skip-gram mode for training.

In the process of using word2vec to train character vector, the Chinese gigaword dataset is used for training, the context scanning window is set to 5 during training, and the dimension of each character vector is 50. After training, the character vector list is obtained. For the fault text, the text input sequence is set to $C = \{c_1, c_2, \ldots, c_n\}$, and the character c_n can find the corresponding character vector x_n in the character vector list. If character does not exist in the list, the character vector is assigned a random value. Finally, the character vector sequence $X = \{x_1, x_2, \ldots, x_n\}$ is generated.

2) STACKED BiLSTM LAYER

In the process of context feature capture, we use stacked BiLSTM to obtain deep context features. Compared with BiLSTM with only one hidden layer to extract features, stacked BiLSTM increases the number of hidden layers to obtain deeper context information. The stacked BiLSTM proposed in this paper is different from the traditional stacked BiLSTM. The output form lower hidden layer is spliced as the input of upper hidden layer. In the traditional stacked BiLSTM, the output from lower hidden layer in the forward direction and the backward direction are used as the input of upper hidden layer in the forward direction and the backward direction, respectively. Figure 5 shows peephole of 3-layer stacked BiLSTM.

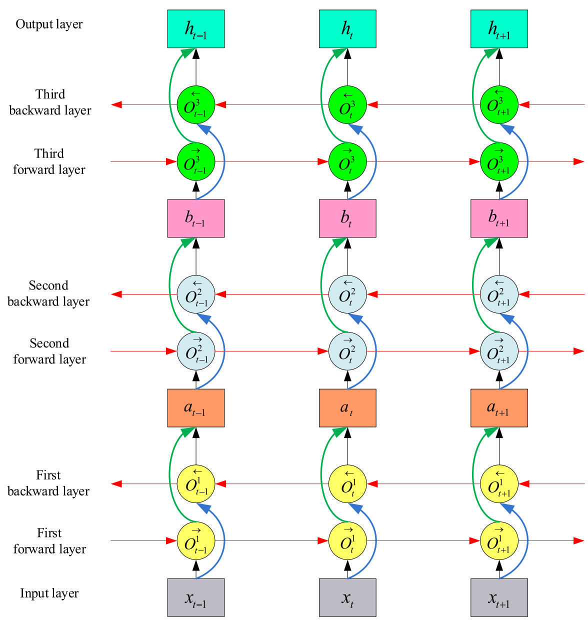


FIGURE 5. Peephole of 3-layer BiLSTM.

As can be seen from figure 5, the nodes of hidden layers represent LSTM units. LSTM unit is composed of input gate, forgetting gate and output gate, denoted as i_t, f_t and o_t, respectively. The input gate indicates which part of the information can be updated to the unit state, the forget gate determines which information in the cell is discarded, and the output gate determines which part of the information to output. g_t represents the unit state at time step t. Given the character vector sequence X, for the first layer BiLSTM, the output of hidden layer a_t for each time step t is given by the following formulas:

$$i_t^{(a)} = \sigma(W_i^{(a)}[O_{t-1}^1, x_t] + b_i^{(a)}) \tag{2}$$

$$f_t^{(a)} = \sigma(W_f^{(a)}[O_{t-1}^1, x_t] + b_f^{(a)}) \tag{3}$$

$$o_t^{(a)} = \sigma(W_o^{(a)}[O_{t-1}^1, x_t] + b_o^{(a)}) \tag{4}$$

$$g_t^{(a)} = \tanh(W_c^{(a)}[O_{t-1}^1, x_t] + b_c^{(a)}) \tag{5}$$

$$c_t^{(a)} = f_t^{(a)} \odot c_{t-1}^{(a)} + i_t^{(a)} \odot g_t^{(a)} \tag{6}$$

$$O_t^1 = o_t^{(a)} \odot \tanh(c_t^{(a)}) \tag{7}$$

$$a_t = [\overrightarrow{O_t^1}, \overleftarrow{O_t^1}] \tag{8}$$

For the second layer BiLSTM, output b_t is given by the following formulas:

$$i_t^{(b)} = \sigma(W_i^{(b)}[O_{t-1}^2, a_t] + b_i^{(b)}) \tag{9}$$

$$f_t^{(b)} = \sigma(W_f^{(b)}[O_{t-1}^2, a_t] + b_f^{(b)}) \tag{10}$$

$$o_t^{(b)} = \sigma(W_o^{(b)}[O_{t-1}^2, a_t] + b_o^{(b)}) \tag{11}$$

$$g_t^{(b)} = \tanh(W_c^{(b)}[O_{t-1}^2, a_t] + b_c^{(b)}) \tag{12}$$

$$c_t^{(b)} = f_t^{(b)} \odot c_{t-1}^{(b)} + i_t^{(b)} \odot g_t^{(b)} \tag{13}$$

$$O_t^2 = o_t^{(b)} \odot \tanh(c_t^{(b)}) \tag{14}$$

$$b_t = [\overrightarrow{O_t^2}, \overleftarrow{O_t^2}] \tag{15}$$

For the third layer BiLSTM, output C_t is given by the following formulas:

$$i_t^{(c)} = \sigma(W_i^{(c)}[O_{t-1}^3, b_t] + b_i^{(c)}) \tag{16}$$

$$f_t^{(c)} = \sigma(W_f^{(c)}[O_{t-1}^3, b_t] + b_f^{(c)}) \tag{17}$$

$$o_t^{(c)} = \sigma(W_o^{(c)}[O_{t-1}^3, b_t] + b_o^{(c)}) \tag{18}$$

$$g_t^{(c)} = \tanh(W_c^{(c)}[O_{t-1}^3, b_t] + b_c^{(c)}) \tag{19}$$

$$c_t^{(c)} = f_t^{(c)} \odot c_{t-1}^{(c)} + i_t^{(c)} \odot g_t^{(c)} \tag{20}$$

$$O_t^3 = o_t^{(c)} \odot \tanh(c_t^{(c)}) \tag{21}$$

$$h_t = [\overrightarrow{O_t^3}, \overleftarrow{O_t^3}] \tag{22}$$

Among them, $\odot$ represents the element-level multiplication calculation, σ represents the sigmod function, and W_i, W_f, W_o, b_i, b_f, and b_o represent the weight matrix and bias term of input gate, forgetting gate and output gate respectively. For the sequence $X = \{x_1, x_2, \ldots, x_n\}$, the deep context feature sequence $H = \{h_1, h_2, \ldots, h_n\}$ is obtained by stacking BiLSTM, which is used as input of event argument entity and relation extraction layer.

3) EVENT ARGUMENT ENTITY AND RELATION EXTRACTION LAYER

In event argument entity and relation extraction layer, the self-attention mechanism is used to obtain character dependency features. On the one hand, it performs EAER. On the other hand, the character dependency features are mapped to the entity label weight embedding, splicing with deep context features to predict event argument relations. This is the first phase of joint extraction in SBALGN.

In EAER task: the self-attention mechanism can obtain the dependency features between any pair of characters, and has been successfully applied in machine translation tasks and labeling tasks. Specifically, after the text sequence is encoded by the stacked BiLSTM, the deep context feature sequence H is obtained. The calculation formula of self-attention mechanism is shown as follows:

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d}})V \tag{23}$$

In formula 23, Q, K, and V correspond to query matrix, keys matrix, and value matrix respectively. $\sqrt{d}$ is an adjustment parameter to prevent the internal product of Q and K from being too large. On this basis, considering that 1-layer attention function cannot obtain the dependency in multiple subspaces, it is necessary to use multi-head attention mechanism. The multi-head attention mechanism maps the input vector to multiple subspaces, and calculates the self-attention mechanism function in the subspaces. The calculation process is repeated N times, and the N times calculation results are spliced to obtain comprehensive feature information. The calculation formula of the multi-head attention mechanism is as follows:

$$M(Q, K, V) = concat(head_1, head_2, \ldots, head_N)W^o \tag{24}$$

$$head_i = Attention(HW_i^Q, HW_i^K, HW_i^V) \tag{25}$$

Among them, W_i^Q, W_i^K, W_i^V and W_o are matrix mapping parameters obtained by training, which are used to map input features into different subspace matrices. *concat* is the splicing operation. Finally, the output of self-attention mechanism is $M = \{m_1, m_2, \ldots, m_n\}$.

In event argument entity category labeling, we use CRF to partition the global optimal labeling sequence. Given the character dependency feature sequence M, the output labeling sequence Y. The joint probability of labeling sequence and feature sequence is as follows:

$$p(M, Y) = \sum_{i=1}^{n} A_{y_{i-1},y_i} + p_{i,y_i} \tag{26}$$

Among them, A represents the transition probability matrix between labels, which can be obtained through training. y_i represents the entity label predicted by the i^{th} character. From formula 27, the conditional probabilities of M and Y can be obtained. In formula 27, Y' represents a possible label sequence and $f(M)$ is the set of all possible labeled sequences. When training CRF, the maximum likelihood estimate is used as the EAER loss function to maximize $p(Y|M)$. The likelihood estimation function is defined as shown in formula 28. By maximizing formula 28, the EAER loss function is obtained, which is defined as formula 29. In the process of event argument entity category prediction, viterbi algorithm is used to predict the optimal event argument entity label sequence.

$$p(Y|M) = \frac{e^{p(M,Y)}}{\sum_{Y' \in f(M)} e^{p(M,Y')}} \tag{27}$$

$$\log(p(Y|M)) = p(M, Y) - \log(\sum_{Y' \in f(M)} e^{p(M,Y')}) \tag{28}$$

$$loss_{ent} = \arg\max_{Y' \in f(M)} \log(p(Y'|M)) \tag{29}$$

In the process of EARE, given the character dependency feature sequence $M = \{m_1, m_2, \ldots, m_n\}$. Since the event argument entity label information of characters can help to predict the event argument relation between entities, the label embedding of common CRF or LSTM decoding result mapping is based on probability inference, and the inference error may affect the event argument relation prediction. In order to solve this problem, this paper introduces a label weight embedding method, which uses the logits output of self-attention mechanism as the input of label weight embedding method. The entity label weight embedding vector of i^{th} character is as follows:

$$e_i = softmax(s(M, i)) \cdot E \tag{30}$$

Among them, $s(M, i)$ is label score function at i^{th} character, and E is label embedding matrix. The *softmax* function enables category labels with high probability to get larger weight, and considers all potential event argument entity labels of character, so as to avoid the problem of CRF prediction error propagation as much as possible.

The output features of stacked BiLSTM are spliced with the label weight embedding vectors to obtain splicing vector z. The event argument relation probability calculation between character c_i and character c_j is defined as shown in formula 31.

$$p(c_i, r, c_j) = \sigma(W_v \tanh(W_f z_{c_i} \oplus W_b z_{c_j})) \tag{31}$$

Among them, $p(c_i, r, c_j)$ represents prediction probability of each event argument relation r between c_i and c_j. $\oplus$ is function of calculating score of c_i and c_j for each relation r. W_v, W_f and W_b are full connected layer weight matrix, forward relation weight matrix and backward relation weight matrix respectively. σ is the *sigmoid* function. It is worth mentioning that the probability $p(c_i, r, c_j)$ is not equal to $p(c_j, r, c_i)$.

In the training process, the cross-entropy loss function is used to obtain minimum optimization goal, as shown in formula 32.

$$loss_{rel} = -\sum_{i=0}^{n-1}\sum_{j=0}^{n-1}\sum_{k=0}^{m-1} q(c_i, r_k, c_j)\log(p(c_i, r_k, c_j)) \tag{32}$$

Among them, n is the number of characters in sentence, and m is the number of relation categories. $q(c_i, r, c_j)$ represents the probability of the relation between the calibration character c_i and character c_j.

4) BiGCN ENTITY RELATION INFERENCE LAYER

The joint extraction of event argument entity and relation in the first phase does not consider the influence of relation on entity recognition. In fact, event argument entity relation categories can better identify entities. For example, the relation category ''Appear'', which makes it easier to recognize entities as ''Status_values'' or ''Equipment''. Because the relation between the characters has been predicted in the first phase, the text sequence can form graph structure. Therefore, this paper introduces BiGCN [58], which takes the deep context features of characters as node features and the character pair relation predicted result in the first phase as adjacency matrix to calculate graph node features. Graph node features are used to update deep context features, and then used for joint extraction of event argument entity and relation in the second phase.

The traditional GCN node connection has no direction, but event argument relation defines head entity and tail entity. According to the orientation of event argument relation, character adjacency matrix is divided into forward adjacency matrix and backward adjacency matrix. The graph convolution formula is as follows:

$$\overrightarrow{h}_u^{l+1} = ReLU(\sum_{v\in V}\sum_{r\in R} p_r(u, v) \cdot (\overrightarrow{W}_r^l h_v^l + \overrightarrow{b}_r^l)) \tag{33}$$

$$\overleftarrow{h}_u^{l+1} = ReLU(\sum_{v\in V}\sum_{r\in R} p_r(v, u) \cdot (\overleftarrow{W}_r^l h_v^l + \overleftarrow{b}_r^l)) \tag{34}$$

$$h_u^{l+1} = [\overrightarrow{h}_u^{l+1}, \overleftarrow{h}_u^{l+1}] \tag{35}$$

Among them, h_u^l is the node feature of node u at layer l. $p_r(u, v)$ represents the probability that the character u and the character v are in the relation r, in $p_r(u, v)$, u is the tail character in the head entity and v is the tail character in the tail entity. $\overrightarrow{W}_r^l$ and $\overrightarrow{b}_r^l$ are forward graph convolution kernel parameters and bias of relation r in layer l, respectively. Similarly, $\overleftarrow{W}_r^l$ and $\overleftarrow{b}_r^l$ are backward graph convolution kernel parameters and bias of relation r in layer l, respectively. V and R are the total number of characters and the total number of relation categories, respectively. In order to improve the accuracy of the two tasks, deep context features are updated according to graph convolution features. The update formula is as follows:

$$h'_u = h_u^{l+1} + h_u \tag{36}$$

Among them, h_u represents the deep context feature of the character u. Finally, The final feature sequence is set to $H' = \{h'_1, h'_2, \ldots, h'_n\}$. Event argument entity and relation joint extraction are performed on the final features again according to the phase in the previous section.

In the whole model training, the total loss function is divided into two kinds, including EAER loss and EARE loss. The loss function of EAER is mainly the sum of maximum likelihood estimate loss function in two phases. Similarly, the loss function of EARE is mainly the sum of cross entropy loss functions of two prediction phases. Finally, the total loss is calculated as the sum of all EAER loss and EARE loss, as defined below:

$$loss_{total} = loss_{1ent} + loss_{2ent} + loss_{1rel} + loss_{2rel} \tag{37}$$

IV. EXPERIMENTS

A. EXPERIMENTAL SETTINGS

1) DATASET

In order to effectively evaluate SBALGN model, this paper collects 900 robot transmission system fault diagnosis event description cases as corpus from the operation and maintenance logs generated by a company's robot transmission

TABLE 1. Statistics of dataset used in our experiments.

Category	Total
Equipment	1504
Sub_equipment	422
Component	863
Part	333
Attribute	653
Attribute_value	646
Status_value	1439
Consist_Of	1307
Lead_To	1099
Appear	2149
Has_Attributes	650

system operation cycle and the reference books written by experts on the Internet [59], [60]. The event description case mainly records the historical fault events of the robot transmission system, as well as the causes of the fault and the corresponding fault phenomenon. We select sentences that describe the fault cause and the fault phenomenon from the case. The average sentence length is 15.01 and the number of characters is 20.7k. The event argument entities in the corpus are labeled. On this basis, the entity relations are further labeled, and a database containing 5860 event argument entities and 5205 semantic relationships is constructed. In order to make better use of the whole dataset, this paper uses 5-fold cross-validation to verify the performance of SBALGN. The information of the corpus is shown in table 1.

2) PARAMETER SETTING AND OPERATING ENVIRONMENT

For input feature vector representation, if there are no additional instructions, we use 50 dimensional character feature vector as the input of all models, all models including SBALGN and other baseline models. Then we developed BiGCN, 1-layer BiLSTM and stacked BiLSTM. These neural networks set the hidden layer dimension to 256. We set dropout rate to 0.9 and the batchsize is fixed to 20. Other network parameters are randomly initialized. In the aspect of performance evaluation, event argument entity and relation joint extraction is considered as correct when both the entity boundary, the entity category and the relation category are correct. The operating environment is set as a 4-core Inter Core i5-7300 processor with a dominant frequency of 3.1 GHz and the memory is 16 GB. Table 2 show SBALGN model parameters.

B. RESULTS

1) OVERALL COMPARISON

In order to test the validity of the model, while avoiding comparison errors caused by different labeling strategies, this paper chooses entity and relation joint extraction methods similar to the labeling strategy in this paper as the baseline methods to compare. All comparative experiments adopt the 5-fold cross validation method. Then the average precision(P), recall(R) and F1 score of EAER, EARE and joint extraction are reported respectively. The comparison results are shown in table 3. The comparison models include:

TABLE 2. Parameters of SBALGN model.

Parameter names	Values
Word vector dimension	50
LSTM hidden unit dimension	256
Label weight embedding dimension	64
BiLSTM stacked layers	5
Dropout rate	0.9
Batchsize	20
Learning rate	0.001

- SPTree [61]: The model recognizes entities through bidirectional sequential LSTM-RNN. The bidirectional treestructured LSTM-RNN is stacked on the bidirectional sequential LSTM-RNN to capture the substructure information of the dependency tree, and is combined with label embedding to identify entity relations.
- Katiyar and Cardie [44]: The method proposes a new stacked BiLSTM based on attention mechanism for entity recognition and relation extraction. Entity recognition label embedding is combined with LSTM decode feature, and the semantic relation between entities can be extracted without accessing dependency tree through attention mechanism.
- Giannis *et al.* [45]: The method proposes a joint extraction neural network, which can be used for entity recognition and relation extraction at the same time, without external natural language processing(NLP) tools and artificial features. Among them, the 3-layer, 64 hidden cells stacked BiLSTM captures context features, CRF layer is used for entity recognition, and relation extraction task is modeled as a multi-head selection problem. The two tasks are related by label embedding.
- Bekoulis *et al.* [62]: The model is similar to the above model, adding adversarial training method to the model input feature representation, and adding small disturbance to the training data.
- Zhang *et al.* [48]: The method proposes a new artificial neural network that uses bidirectional LSTM module to obtain entity context information for entity recognition. The entity labels and context information in the process of entity recognition are further transferred to CNN network for relation classification.
- Huang *et al.* [63]: The method introduces bidirectional encoder representation from transformers(BERT) into entity and entity relation joint extraction, uses semantically enhanced BERT as feature extraction layer to obtain context features. Soft label embedding technology is proposed to transfer the information between entity recognition and relation extraction. Finally, global relation prediction is proposed to guide the feature learning process.

It can be seen from the results in table 3, compared with SPTree, it shows 1.04% and 1.28% improvement advantages in EAER and EARE tasks, respectively. The reason for this difference is that the SPTree model relies on the shortest dependency tree structure between entity pairs marked by

NLP tool, and the dependency tree structure is may not necessarily accurate for different languages. SBALGN uses label weight embedding and relation inference as the connection between subtasks, and combines stacked BiLSTM and self-attention mechanism to further improve the performance of entity and relation joint extraction.

Compared with Katiyar and Cardie [44], it shows 2.15% advantage in joint extraction performance. This is because SBALGN uses the sigmoid function to identify multiple relations and solves the problem of overlapping entity relations in practice. In addition, SBALGN uses self-attention mechanism to obtain useful information in text and improve the feature representation of text. CRF is used to decode features which can obtain the dependency relation between entity labels. BiGCN relation inference is used to infer and correct the wrong entities and relations in the second phase, so as to improve the accuracy of SBALGN.

Compared with Bekoulis *et al.* [45], the performance of SBALGN is improved by 0.92% in joint extraction, and improves significantly on EARE tasks, reaching 1.27%. It shows that the label weight embedding proposed in this paper can avoid the false prediction of CRF labels into EARE task to some extent. Secondly, the BiGCN relation inference proposed in this paper can improve the accuracy of EAER tasks, and then improve the performance of EARE. In addition, self-attention mechanism is also one of the factors that improve EARE performance. Compared with Giannis *et al.* [62], although the model adds adversarial training to the input representation, the performance of joint extraction is still 0.74% lower than that of SBALGN.

Compared with Zhang *et al.* [48], it improves 1.17% in entity recognition task and 1.42% in joint extraction. It shows that the introduction of self-attention mechanism in this paper can further capture the structural information of text sequence and improve the performance of EAER. The use of stacked BiLSTM can obtain deep context features to understand sentence structure, which is beneficial to improve the accuracy of EARE. Enhancing the interconnection between the two subtasks through label weight embedding and BiGCN relation inference methods is also one of the reasons to improve the performance of joint extraction.

In addition, SBALGN needs more training time than the above-mentioned methods. This is because SBALGN emphasizes the key features of text acquisition by stacked BiLSTM and self-attention mechanism, and the two phase joint extraction improves the correlation between subtasks. Compared with the one phase joint extraction methods, the time cost increases, but the performance improvement is more obvious.

Compared with Huang *et al.* [63], the model proposed in this paper reduces 3.69% in EAER and improves 3.22% in EARE, which proves that the stacked BiLSTM proposed in this paper can obtain the deep context information of text and improve the accuracy of EARE, the two phase relation extraction can further improve the performance of EARE. Overall, the F1 value of joint extraction is slightly reduced by 0.23%. However, the method of Huang *et al.* [63] uses the BERT pre-trained model to generate character feature vectors, and improves the representation ability of character features by adding training parameters. The cost of increasing training parameters is to spend a lot of training time. The word2vec used in this paper does not need to dynamically generate character features, the training time required is greatly reduced, and the average iteration time is reduced from 990.63s to 68.74s. Therefore, when the performance gap is small, considering the calculation cost, the performance of Huang *et al.* [63] is still not as good as SBALGN.

2) EFFECT OF EACH COMPONENT OF SBALGN

SBALGN consists of four parts, including stacked BiLSTM, self-attention mechanism, BiGCN relation inference and label weight embedding. The experiment in this section mainly prove that the four components all contribute to the final results and the contribution of each component on the performance of SBALGN.

In the experiment, SBALGN is divided into the following 5 situations: baseline indicates that SBALGN only uses 1-layer BiLSTM, removes the self-attention mechanism, label weight embedding and BiGCN. baseline+stacked BiLSTM indicates that the baseline method adds stacked BiLSTM, baseline+stacked BiLSTM + self-attention indicates that the baseline method adds stacked BiLSTM and self-attention mechanism, baseline+stacked BiLSTM + self-attention+label weight embedding indicates that the baseline method increases stacked BiLSTM, self-attention mechanism and label weight embedding. These four models are compared with the SBALGN model including all components. Similar to the experiment in the previous section, the number of stacked layers is set to 5, the number of hidden units is set to 256, the batchsize is set to 20, and other parameters remain unchanged. 5-fold cross-validation method is used to verify the performance of the model. The results are shown in table 4.

As can be seen from the results in table 4, compared with baseline, the joint extraction performance of baseline+stacked BiLSTM is improved by 1.34%, and 1.92% in EARE subtask. It has been proved that the stacked BiLSTM can obtain the deep context features of sequences, which can greatly improve the performance of joint extraction. Compared with the baseline+stacked BiLSTM, the baseline+stacked BiLSTM+self-attention method adds self-attention mechanism, which improves 0.2% in EAER part, which indicates that self-attention mechanism can further obtain dependency from subspace and improve entity recognition performance in the first phase of joint extraction. Compared with baseline+stacked BiLSTM+self-attention, baseline+stacked BiLSTM+self-attention+label weight embedding method is improved by 0.51% in EARE, which indicates that adding label weight embedding can improve EARE accuracy. Compared with baseline+stacked BiLSTM+self-attention+label weight embedding, the second phase of joint extraction proposed in this paper can add relation information to EAER and improve the performance

TABLE 3. Comparison of our method with the state-of-the-art on our event argument entity and relation joint extraction dataset.

Model	EAER			EARE			Joint F1 score	Average training time(ep)/s
	P	R	F1 score	P	R	F1 score		
SPTree [61]	91.25%	92.28%	91.76%	82.11%	81.04%	81.57%	86.67%	28.09
Katiyar et al [44]	90.84%	91.15%	90.99%	81.00%	79.74%	80.37%	85.68%	23.97
Giannis et al [45]	92.07%	92.43%	92.25%	81.55%	81.62%	81.58%	86.91%	19.93
Giannis et al [62]	91.89%	92.62%	92.25%	81.74%	82.11%	81.92%	87.09%	20.81
Zhang et al [48]	91.32%	91.94%	91.63%	82.55%	79.87%	81.19%	86.41%	16.49
Huang et al [63]	**96.25%**	**96.71%**	**96.49%**	80.44%	78.83%	79.63%	**88.06%**	990.63
SBALGN(ours)	92.60%	93.01%	92.80%	**83.25%**	**82.45%**	**82.85%**	87.83%	68.74

TABLE 4. Effect of each component on the performance of SBALGN.

Model	EAER			EARE			Joint F1 score
	P	R	F1 score	P	R	F1 score	
Baseline	90.79%	91.97%	91.38%	81.52%	78.64%	80.05%	85.72%
Baseline+stacked BiLSTM	91.70%	92.60%	92.15%	82.18%	81.77%	81.97%	87.06%
Baseline+stacked BiLSTM+self-attention	91.95%	92.75%	92.35%	82.14%	81.98%	82.06%	87.21%
Baseline+stacked BiLSTM+self-attention+label weight embedding	91.84%	92.78%	92.31%	83.15%	82.01%	82.57%	87.44%
SBALGN(ours)	**92.60%**	**93.01%**	**92.80%**	**83.25%**	**82.45%**	**82.85%**	**87.83%**

of EAER. At the same time, the second phase of joint extraction can correct the misidentified triples in the first phase, and improve the performance of EAER and EARE by 0.49% and 0.28%, respectively. Finally, compared with the baseline method, SBALGN has the best effect when four components are included, which verifies the effectiveness of all components for SBALGN performance.

3) TUNING OF HYPERPARAMETERS IN SBALGN

In SBALGN, stacked BiLSTM is used to obtain deep text features. Different stack layers have different effects on the model performance, and the number of LSTM hidden cells also has an impact on model performance. Therefore, it is particularly important to select the appropriate number of stacked layers and hidden cells. In order to evaluate the influence of the above two factors on the performance of the model, a set of experiments will be conducted to study the influence of the two parameters on the joint extraction. The number of stacked layers m is as follows: $m = 1, 2, 3, 4, 5$. The number of LSTM hidden cells n is as follows: $n = 32, 64, 128, 256$. All other parameters remain unchanged and 5-fold cross-validation is applied to the model. The result is shown in table 5. In addition, we also compared the influence of the selection of different number of hidden cells on the average loss of training set, the accuracy of training set and the accuracy of validation set. The number of stacked BiLSTM layers is set to 5 and other parameters remain unchanged. The comparison results are shown in figure 6. The accuracy calculation formula is shown in formula 38, among them, TP represents true positive, TN represents true negative, FP represents false positive, and FN represents false negative.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{38}$$

As shown in table 5, when the number of stacked BiLSTM layers is fixed, except when the number of stacked BiLSTM layers is set to 2, increasing the number of LSTM hidden cells will reduce the performance of SBALGN. In other situations, it can be seen that more LSTM hidden cells can bring better performance to SBALGN. This result can prove that the more number of BiLSTM hidden cells, the richer text context information can be obtained, which is very important for small-scale sample datasets. When the number of LSTM hidden cells is fixed, increasing the number of stacked BiLSTM layers can improve the performance of entity recognition and relation extraction. Except that the number of LSTM hidden cells is set to 32 and 64, the number of stacked layers is larger than 2, the performance of SBALGN decreases. The results show that when more hidden cells are fixed, stacking more layers can help to obtain more context features and improve performance.

It can be seen from figure 6 that more LSTM hidden layer cells can bring better performance. For example, the accuracy rate of the verification set of 256 LSTM hidden layer cells is higher than that of 128 LSTM hidden layer cells. In addition, the model with 256 LSTM hidden layer cells has a faster convergence rate in training average loss and accuracy than the other three models. In summary, this paper adopts 5-layer stacked BiLSTM and 256 hidden cells as the optimal parameters of SBALGN.

In SBALGN training, the selection of batchsize also affects the performance of joint extraction model. Batchsize refers to randomly dividing the dataset into several batches. The sample set of each batch contains batchsize samples. When training, batchsize training samples are used to calculate the gradient and update the model parameters. We set up a set of experiments to verify the effect of batchsize on the performance of joint extraction. The value of batchsize is set to $b = 1, 10, 20, 30$, the number of stacked BiLSTM layers is set to 5, the number of LSTM hidden cells is set to 256, and other parameters remain unchanged. Perform 5-fold cross validation on the dataset and report the average results. The results are shown in table 6. Similarly, we compared the

TABLE 5. Impact of stacked BiLSTM layers number and hidden cells number.

Number of LSTM hidden cells	EAER P	EAER R	EAER F1 score	EARE P	EARE R	EARE F1 score	Joint F1 score
No. of Stacked BiLSTM Layers: 1							
32	91.16%	91.44%	91.30%	78.85%	77.97%	78.41%	84.86%
64	91.21%	92.51%	91.86%	81.30%	79.59%	80.44%	86.15%
128	91.58%	92.46%	92.02%	80.79%	80.09%	80.44%	86.23%
256	**91.50%**	**92.37%**	**91.93%**	**82.32%**	**78.95%**	**80.60%**	**86.27%**
No. of Stacked BiLSTM Layers: 2							
32	92.06%	92.42%	92.24%	81.31%	80.86%	81.08%	86.66%
64	**92.22%**	**92.82%**	**92.52%**	**82.78%**	**81.74%**	**82.26%**	**87.39%**
128	92.27%	92.62%	92.44%	83.11%	81.14%	82.11%	87.28%
256	91.90%	92.70%	92.30%	82.65%	81.22%	81.93%	87.12%
No. of Stacked BiLSTM Layers: 3							
32	91.35%	92.03%	91.69%	81.25%	80.91%	81.08%	86.39%
64	92.07%	92.63%	92.35%	82.56%	81.50%	82.03%	87.19%
128	91.97%	92.99%	92.48%	82.70%	81.62%	82.16%	87.32%
256	**92.23%**	**92.89%**	**92.56%**	**82.77%**	**81.87%**	**82.32%**	**87.44%**
No. of Stacked BiLSTM Layers: 4							
32	91.32%	91.91%	91.61%	81.13%	80.39%	80.76%	86.19%
64	91.74%	92.51%	92.12%	82.31%	81.46%	81.88%	87.00%
128	91.96%	92.85%	92.40%	82.56%	81.87%	82.21%	87.31%
256	**92.15%**	**92.94%**	**92.54%**	**82.73%**	**81.97%**	**82.35%**	**87.45%**
No. of Stacked BiLSTM Layers: 5							
32	91.21%	91.98%	91.59%	81.30%	80.95%	81.12%	86.36%
64	91.62%	92.50%	92.06%	82.26%	81.58%	81.92%	86.99%
128	92.07%	92.76%	92.41%	83.15%	81.92%	82.53%	87.47%
256	**92.60%**	**93.01%**	**92.80%**	**83.25%**	**82.45%**	**82.85%**	**87.83%**

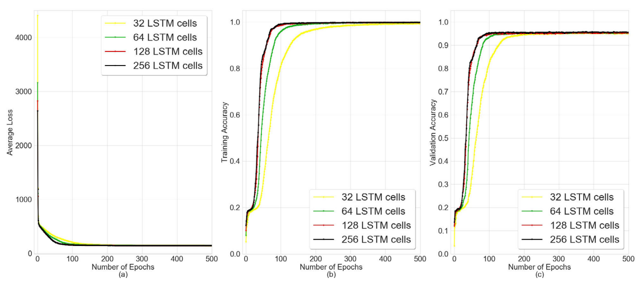


FIGURE 6. Performance of stacked BiLSTM with different number of hidden cells. (a) Average loss. (b) Training accuracy. (c) Validation accuracy.

influence of different batchsize values on the average training loss, training set accuracy and verification set accuracy. The results are shown in figure 7.

It can be seen from table 6 that the performance of joint extraction improves with the increase of batchsize, which shows that selecting appropriate number of samples to form batch training can not only speed up the training speed, but also can calculate the gradient more accurately and obtain better results. When the value of batchsize is larger than 20, the performance of the joint extraction model decreases, which indicates that the batchsize is too large to easily converge the gradient to the bad local optimal value, and has poor generalization ability.

It can be seen from figure 7 that increasing the batchsize brings higher accuracy to the verification set. For example, the verification accuracy rate with 20 batchsize is higher than that with 10 batchsize. In terms of the average loss of the training set, the larger value with batchsize, the higher with average loss. This is because zero marks are added to fix the length of batch samples in batch training. Although in the training process, the convergence speed is the fastest when the value of batchsize is 1, the stability of network training is poor, and the accuracy curve appears to oscillate. Considering the extraction accuracy, it is more reasonable to choose smaller batchsize. In summary, when the value of batchsize is set to 20, the performance of the model is better.

In the model training, in order to make better use of limited text data, this paper adopts k-fold cross-validation method for method verification. The number of k-folds in cross-validation also affects the performance of SBALGN. This experiments mainly verify the influence of different k-fold numbers on the performance of joint extraction. We fixed the number of stacked BiLSTM layers as 5, the number of LSTM hidden cells as 256, the number of batchsize as 20, and other parameters remain unchanged. The experimental results are shown in table 7

TABLE 6. Effect of batchsize value on the performance of SBALGN.

Batchsize	EAER			EARE			Joint F1 score
	P	R	F1 score	P	R	F1 score	
1	92.03%	92.60%	92.31%	82.33%	81.57%	81.95%	87.13%
10	92.36%	92.94%	92.65%	82.55%	82.62%	82.58%	87.62%
20	**92.60%**	**93.01%**	**92.80%**	**83.25%**	**82.45%**	**82.85%**	**87.83%**
30	91.97%	92.78%	92.37%	83.10%	82.40%	82.75%	87.56%

TABLE 7. Comparison of k-fold numbers in cross-validation.

No. of k-fold in cross-validation	EAER			EARE			Joint F1 score
	P	R	F1 score	P	R	F1 score	
3	91.09%	92.05%	91.57%	81.35%	80.72%	81.03%	86.30%
4	92.08%	92.79%	92.43%	82.46%	82.07%	82.26%	87.35%
5	**92.60%**	**93.01%**	**92.80%**	**83.25%**	**82.45%**	**82.85%**	**87.83%**
6	92.13%	92.87%	92.50%	83.08%	82.34%	82.71%	87.61%

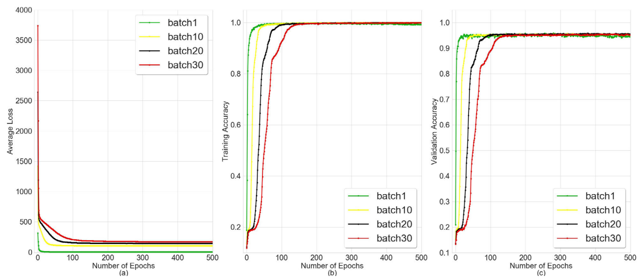


FIGURE 7. Performance of batchsize with different value. (a) Average loss. (b) Training accuracy. (c) Validation accuracy.

As can be seen from table 7 that compared with 4-fold, 5-fold, and 6-fold cross-validation methods, 3-fold cross-validation has the lowest performance. This is because the number of samples involved in the training of 3-fold cross-validation is less than other methods, which cannot reflect the effective information of data, so the performance is the lowest. As the number of cross-validation fold increases, the joint extraction performance is also improved. When performing 6-fold cross-validation, the joint extraction performance decreases. This is because the more data put into the training set, the smaller the deviation of the model, which leads to the larger variance in the error rate of the validation set, leading to overfitting. Therefore, 5-fold cross-validation is the optimal cross-validation fold.

4) COMPARISON OF SBALGN JOINT EXTRACTION PHASES AND BiGCN LAYERS

In SBALGN, BiGCN is used to infer the result of event argument relation prediction. In section 4.2.2, it has been proved that event argument relation category information is beneficial to improve the accuracy of EAER and also improve the performance of EARE in the second phase. In order to verify the optimal setting of two phases joint extraction and 1-layer BiGCN inference. We study the joint extraction results of multiple phases and multi-layer BiGCN, where the number of stacked BiLSTM layer is set to 5, the number of LSTM hidden cells is set to 256 and batchsize is set to 20. The average results of 5-fold cross validation are shown in table 8.

It can be seen from table 8 that stacking more BiGCN layers not only increases the model training parameters, but also cannot improve the joint extraction performance. In SBALGN, the setting of 1-layer BiGCN should be the most suitable. We also added the third phase of joint extraction. The third phase of joint extraction is mainly based on the results of the second phase of joint extraction through BiGCN update features, again for event argument entity and relation joint extraction. The results show that the joint extraction in the third phase reduces the performance, because the joint extraction error results of previous phases are easy to accumulate in the third phase. Therefore, 2nd-phase is sufficient for joint extraction in this paper.

5) COMPARISON OF ENTITY RELATIONS EXTRACTION FUNCTIONS

In the robot transmission system fault diagnosis text, there are situations where one entity involves multiple relations. In SBALGN, the sigmoid function is used to predict the relation between entities. The sigmoid function can independently consider relation category, and the probability all categories is not necessarily to sum up to 1. This paper compares the softmax function used in relation prediction in reference [43]. The model parameter batchsize is fixed to 20, the number of LSTM hidden cells is 256, the number of stacked BiLSTM layers is set to 5, and other parameters remain unchanged. The experimental results are shown in table 9.

It can be seen from table 9 that the use of softmax function in relation prediction tasks decreases significantly. This is because softmax mainly predicts the relation category with the largest prediction probability and cannot solve the problem of overlapping entity relation prediction. The sigmoid can independently predict the relation category, which can solve the problem that one entity involves multiple entity relations.

6) VISUALIZATION OF EVENT LOGIC KNOWLEDGE GRAPH

The knowledge triples obtained by the above-mentioned event argument entity and relation joint extraction method

TABLE 8. Impact of BiGCN layers number and SBALGN phase number.

Phase	BiGCN layers	EAER			EARE			Joint F1 score
		P	R	F1 score	P	R	F1 score	
2nd-phase	**1**	**92.60%**	**93.01%**	**92.80%**	**83.25%**	**82.45%**	**82.85%**	**87.83%**
2nd-phase	2	92.05%	92.58%	92.31%	82.78%	82.37%	82.57%	87.44%
3rd-phase	1	92.19%	92.82%	92.50%	83.36%	81.84%	82.59%	87.55%

TABLE 9. Comparison of relation prediction methods.

Method	EAER			EARE			Joint F1 score
	P	R	F1 score	P	R	F1 score	
Softamx	91.63%	92.46%	92.04%	82.99%	74.37%	78.44%	85.24%
Sigmoid	**92.60%**	**93.01%**	**92.80%**	**83.25%**	**82.45%**	**82.85%**	**87.83%**

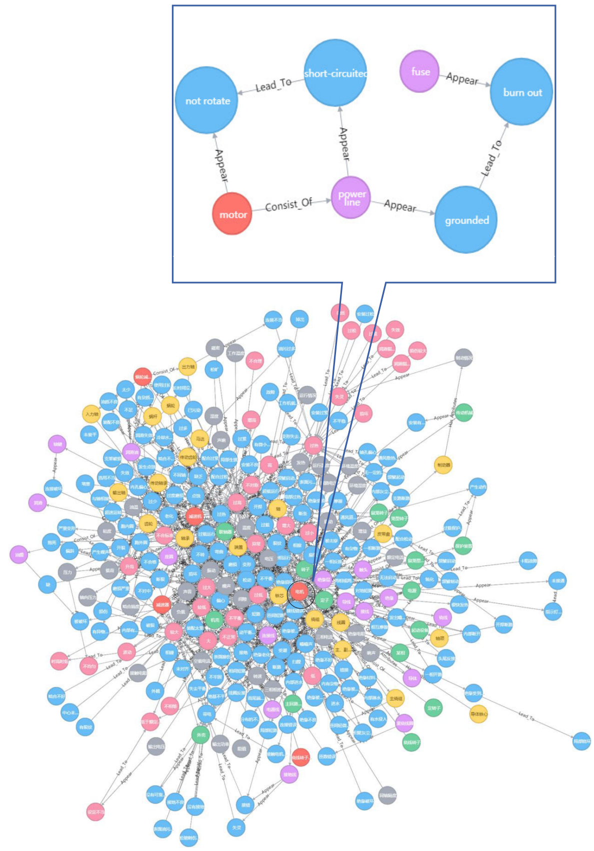


FIGURE 8. Visualization of event logic knowledge graph of robot transmission system fault diagnosis(partial).

need to be stored in a specific physical structure. This paper uses a graph database for storage. In the graph database management system, neo4j is currently one of the most popular graph database software, which mainly stores graph structure data with nodes and relations as objects. Therefore, this paper uses neo4j to store the fault diagnosis event logic knowledge tuples of robot transmission system, and initially establishes a fault diagnosis event logic knowledge graph. Figure 8 shows part of the event logic knowledge graph, and the detail graph is an example English translation of entity and relation triples. The nodes corresponding to similar entities are displayed in uniform color, for example, the color of the node corresponding to the state_value type entity is blue, and the color of the node corresponding to the equipment type entity is red. From the detailed graph in figure 8, we can see a certain fault phenomenon of motor and corresponding fault cause cases. Among them, the motor contains power line parts, when the power line is short-circuited or grounded, the fuse will burn out and the motor will not rotate. The fault phenomenon and cause can be clearly displayed.

V. CONCLUSION

The construction of fault diagnosis event logic knowledge graph of robot transmission system is an important technology to realize automatic knowledge management. In this paper, a top-down event logic knowledge graph method is proposed based on the historical fault diagnosis event description text of robot transmission system. Firstly, a fault diagnosis event logic knowledge ontology is constructed, then event argument entity and relation labeling strategy is performed according to ontology model and form an event argument entity and relation joint extraction dataset. A new event argument entity and relation joint extraction model is proposed, which does not rely on natural language processing tools. It uses stacked BiLSTM and self-attention mechanism to obtain multi-level features and character dependency features in text itself. Then entity label weight embedding vector and BiGCN relation inference method enhancer the interconnection of subtasks is proposed. Experiments on the corpus of robot transmission system fault diagnosis show that the model has better performance than the latest entity and relation joint extraction models in recent years. Finally, a preliminary fault diagnosis event logic knowledge graph is established to provide decision support for knowledge diagnosis.

Although we have initially realized the construction of fault diagnosis event logic knowledge graph, there are still limitations. In the next step, the first is to expand the corpus, mainly to expand the corpus of robot fault resolution, equipment operation and maintenance, then to expand the ontology of event logic knowledge graph. The second is to improve the input feature representation of the event argument entity and relation joint extraction model, such as multi-granular character feature representation, so as to further improve the performance of joint extraction. Finally, conduct extended study on knowledge fusion and knowledge inference to process the knowledge fusion in the event logic knowledge graph.

REFERENCES

[1] Z. Wei, W. Kai, X. Aidong, Z. Peng, Y. Shunkun, S. Yue, and G. Haifeng, "An industrial robot health assessment method for intelligent manufacturing," *Robot*, vol. 42, no. 4, pp. 460–468, 2020.

[2] Y. Hong, Z. Sun, X. Zou, and J. Long, "Multi-joint industrial robot fault identification using deep sparse auto-encoder network with attitude data," in *Proc. Prognostics Health Manage. Conf. (PHM-Besançon)*, 2020, pp. 176–179.

[3] L. Chen, J. Cao, and X. Wang, "Nonlinear spectrum based fault diagnosis method for industrial robot drive systems," *J. Xi'an Jiaotong Univ.*, vol. 53, no. 4, pp. 93–99, 2019.

[4] H.-K. Hsu, H.-Y. Ting, M.-B. Huang, and H.-P. Huang, "Intelligent fault detection, diagnosis and health evaluation for industrial robots," *Mechanics*, vol. 27, no. 1, pp. 70–79, Feb. 2021.

[5] Y. Zhang, X. Qi, and Y. Tong, "Fault diagnosis of welding robot based on expert system," *Mach. Tool Hydraul.*, vol. 47, no. 1, pp. 181–186, 2019.

[6] M. Castellano-Quero, J.-A. Fernandez-Madrigal, and A. J. Garcia-Cerezo, "Integrating multiple sources of knowledge for the intelligent detection of anomalous sensory data in a mobile robot," in *Proc. 4th Iberian Robot. Conf. (Robot)-Adv. Robot.* Cham, Switzerland: Springer, 2019, pp. 159–170.

[7] D. Zhang, Z. Liu, W. Jia, H. Liu, and J. Tan, "A review on knowledge graph and its application prospects to intelligent manufacturing," *J. Mech. Eng.*, vol. 57, no. 5, pp. 90–113, 2021.

[8] Q. Wang, Z. Mao, and B. Wang, "Knowledge graph embedding: A survey of approaches and applications," *IEEE Trans. Knowl. Data Eng.*, vol. 29, no. 12, pp. 2724–2743, Dec. 2017.

[9] T. Ebisu and R. Ichise, "Generalized translation-based embedding of knowledge graph," *IEEE Trans. Knowl. Data Eng.*, vol. 32, no. 5, pp. 941–951, May 2020.

[10] S. Guo, Q. Wang, B. Wang, L. Wang, and L. Guo, "SSE: Semantically smooth embedding for knowledge graphs," *IEEE Trans. Knowl. Data Eng.*, vol. 29, no. 4, pp. 884–897, Apr. 2017.

[11] X. Chen, S. Jia, and Y. Xiang, "A review: Knowledge reasoning over knowledge graph," *Expert Syst. Appl.*, vol. 141, Mar. 2020, Art. no. 112948.

[12] R. Guo, Q. Yang, S. Liu, W. Li, X. Yuan, and X. Huang, "Construction and application of power grid fault handing knowledge graph," *Power Syst. Technol.*, vol. 45, no. 6, pp. 2092–2100, 2021.

[13] W. Zhang, J. Wang, Y. Wang, and Z. Sun, "Constructing and inferring event logic cognitive graph in the field of big data," *Scientia Sinica Informationis*, vol. 50, no. 7, pp. 988–1002, Jul. 2020.

[14] J. Feng, Y. Yao, S. Lu, and Y. Liu, "Domain knowledge-based deep-broad learning framework for fault diagnosis," *IEEE Trans. Ind. Electron.*, vol. 68, no. 4, pp. 3454–3464, Apr. 2021.

[15] J. Wang and J. Lv, "Tag-informed collaborative topic modeling for cross domain recommendations," *Knowl.-Based Syst.*, vol. 203, Sep. 2020, Art. no. 106119.

[16] C. Bizer, J. Lehmann, G. Kobilarov, S. Auer, C. Becker, R. Cyganiak, and S. Hellmann, "DBpedia—A crystallization point for the web of data," *J. Web Semantics*, vol. 7, no. 3, pp. 154–165, Sep. 2009.

[17] J. Hoffart, F. M. Suchanek, K. Berberich, and G. Weikum, "YAGO2: A spatially and temporally enhanced knowledge base from Wikipedia," *Artif. Intell.*, vol. 194, pp. 28–61, Jan. 2013.

[18] D. Vrandečić and M. Krötzsch, "Wikidata: A free collaborative knowledgebase," *Commun. ACM*, vol. 57, no. 10, pp. 78–85, 2014.

[19] B. Abu-Salih, "Domain-specific knowledge graphs: A survey," *J. Netw. Comput. Appl.*, vol. 185, Jul. 2021, Art. no. 103076.

[20] K. M. Malik, M. Krishnamurthy, M. Alobaidi, M. Hussain, F. Alam, and G. Malik, "Automated domain-specific healthcare knowledge graph curation framework: Subarachnoid hemorrhage as phenotype," *Expert Syst. Appl.*, vol. 145, May 2020, Art. no. 113120.

[21] T. Ruan, L. Xue, H. Wang, F. Hu, L. Zhao, and J. Ding, "Building and exploring an enterprise knowledge graph for investment analysis," in *Proc. Int. Semantic Web Conf.* Cham, Switzerland: Springer, 2016, pp. 418–436.

[22] D. Shi, T. Wang, H. Xing, and H. Xu, "A learning path recommendation model based on a multidimensional knowledge graph framework for e-learning," *Knowl.-Based Syst.*, vol. 195, May 2020, Art. no. 105618.

[23] H. L. Nguyen and J. J. Jung, "Social event decomposition for constructing knowledge graph," *Future Gener. Comput. Syst.*, vol. 100, pp. 10–18, Nov. 2019.

[24] K. Guo, T. Jiang, and H. Zhang, "Knowledge graph enhanced event extraction in financial documents," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Dec. 2020, pp. 1322–1329.

[25] M. Ringsquandl, E. Kharlamov, D. Stepanova, S. Lamparter, R. Lepratti, I. Horrocks, and P. Kroger, "On event-driven knowledge graph completion in digital factories," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Dec. 2017, pp. 1676–1681.

[26] M. Rospocher, M. van Erp, P. Vossen, A. Fokkens, I. Aldabe, G. Rigau, A. Soroa, T. Ploeger, and T. Bogaard, "Building event-centric knowledge graphs from news," *J. Web Semantics*, vols. 37–38, pp. 132–151, Mar. 2016.

[27] Z. Li, X. Ding, and T. Liu, "Constructing narrative event evolutionary graph for script event prediction," in *Proc. 27th Int. Joint Conf. Artif. Intell.*, Jul. 2018.

[28] C. Li, G. Tian, M. Zhang, and Y. Zhang, "Ontology-based humanoid cognition and reasoning of object attributes," *J. Zhejiang Univ.*, vol. 52, no. 339, pp. 6–13, 2018.

[29] L. Zhou, C. Zhang, I. A. Karimi, and M. Kraft, "An ontology framework towards decentralized information management for eco-industrial parks," *Comput. Chem. Eng.*, vol. 118, no. 4, pp. 49–63, Oct. 2018.

[30] J. R. Martínez-García, F.-E. Castillo-Barrera, R. R. Palacio, G. Borrego, and J. C. Cuevas-Tello, "An Ontology for knowledge condensation to support expertise location in the code phase during software development process," *IET Softw.*, vol. 14, no. 3, pp. 234–241, Jun. 2020.

[31] H. Guan and D. Xie, "Research on building of acupuncture domain ontology," in *Proc. IEEE 19th Int. Conf. e-Health Netw., Appl. Services (Healthcom)*, Oct. 2017, pp. 1–4.

[32] W. Fang, Y. Li, Y. Xiong, J. Chen, and X. Yan, "Research on semantic retrieval for communication ontology," in *Proc. Int. Conf. Intell. Comput. Technol. Automat.*, 2016, pp. 756–760.

[33] W. Zhong, C. Li, X. Peng, F. Wan, X. An, and Z. Tian, "A knowledge base system for operation optimization: Design and implementation practice for the polyethylene process," *Engineering*, vol. 5, no. 6, pp. 1041–1048, Dec. 2019.

[34] Q. Zhou, P. Yan, H. Liu, and Y. Xin, "A hybrid fault diagnosis method for mechanical components based on ontology and signal analysis," *J. Intell. Manuf.*, vol. 30, no. 4, pp. 1693–1715, Apr. 2019.

[35] D. Geng and X. Fu, "Research on fault diagnosis mechanism of production line equipment based on semantic," in *Proc. IEEE 10th Int. Conf. Electron. Inf. Emergency Commun. (ICEIEC)*, Jul. 2020, pp. 220–223.

[36] Z. Wang, Y. Qian, L. Wang, S. Zhang, and X. Luo, "The extraction of hidden fault diagnostic knowledge in equipment technology manual based on semantic annotation," in *Proc. 8th Int. Conf. Softw. Comput. Appl.*, Feb. 2019, pp. 419–424.

[37] Q. Zhou, P. Yan, H. Liu, Y. Xin, and Y. Chen, "Research on a configurable method for fault diagnosis knowledge of machine tools and its application," *Int. J. Adv. Manuf. Technol.*, vol. 95, nos. 1–4, pp. 937–960, Mar. 2018.

[38] Q. Zhou, P. Yan, and Y. Xin, "Research on a knowledge modelling methodology for fault diagnosis of machine tools based on formal semantics," *Adv. Eng. Informat.*, vol. 32, pp. 92–112, Apr. 2017.

[39] F. Xu, X. Liu, W. Chen, C. Zhou, and B. Cao, "Ontology-based method for fault diagnosis of loaders," *Sensors*, vol. 18, no. 3, p. 729, Feb. 2018.

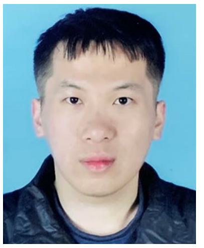
[40] E. Tsalapati, C. W. D. Johnson, T. W. Jackson, L. Jackson, D. Low, B. Davies, L. Mao, and A. West, "Enhancing polymer electrolyte membrane fuel cell system diagnostics through semantic modelling," *Expert Syst. Appl.*, vol. 163, Jan. 2021, Art. no. 113550.

[41] D. L. Nuñez and M. Borsato, "OntoProg: An ontology-based model for implementing prognostics health management in mechanical machines," *Adv. Eng. Informat.*, vol. 38, pp. 746–759, Oct. 2018.

[42] H. Liu, P. Yan, Q. Zhou, Y. Xin, Z. Zhang, and J. Pan, "Machine tool fault diagnosis knowledge expansion method based on semantic web," *Comput. Integr. Manuf. Syst.*, vol. 26, no. 3, pp. 49–62, 2020.

[43] S. Zheng, F. Wang, H. Bao, Y. Hao, P. Zhou, and B. Xu, "Joint extraction of entities and relations based on a novel tagging scheme," in *Proc. 55th Annu. Meeting Assoc. Comput. Linguistics*, vol. 1, 2017, pp. 1227–1236.

[44] A. Katiyar and C. Cardie, "Going out on a limb: Joint extraction of entity mentions and relations without dependency trees," in *Proc. 55th Annu. Meeting Assoc. Comput. Linguistics*, vol. 1, 2017, pp. 917–928.
[45] G. Bekoulis, J. Deleu, T. Demeester, and C. Develder, "Joint entity recognition and relation extraction as a multi-head selection problem," *Expert Syst. Appl.*, vol. 114, pp. 34–45, Dec. 2018.
[46] T.-J. Fu and W.-Y. Ma, "Graphrel: Modeling text as relational graphs for joint entity and relation extraction," in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*, 2019, pp. 1409–1418.
[47] Y. Zhou, L. Huang, T. Guo, S. Hu, and J. Han, "An attention-based model for joint extraction of entities and relations with implicit entity features," in *Proc. Companion World Wide Web Conf.*, 2019, pp. 729–737.
[48] Z. Zhang, S. Zhan, H. Zhang, and X. Li, "Joint model of entity recognition and relation extraction based on artificial neural network," *J. Ambient Intell. Hum. Comput.*, vol. 4, pp. 1–9, 2020.
[49] M. Cao, Z. Yang, L. Luo, H. Lin, and J. Wang, "Joint drug entities and relations extraction based on neural networks," *J. Comput. Res. Develop.*, vol. 56, no. 7, pp. 1432–1440, 2019.
[50] Z. Cui, R. Ke, Z. Pu, and Y. Wang, "Deep bidirectional and unidirectional LSTM recurrent neural network for network-wide traffic speed prediction," 2018, *arXiv:1810.02143*.
[51] J. Zhou, Y. Lu, H.-N. Dai, H. Wang, and H. Xiao, "Sentiment analysis of Chinese microblog based on stacked bidirectional LSTM," *IEEE Access*, vol. 7, pp. 38856–38866, 2019.
[52] X. Zeng, D. Zeng, S. He, K. Liu, and J. Zhao, "Extracting relational facts by an end-to-end neural model with copy mechanism," in *Proc. 56th Annu. Meeting Assoc. Comput. Linguistics*, vol. 1, Jul. 2018, pp. 506–514.
[53] J. Tian, X. Lv, X. You, G. Xiao, and J. Han, "Joint extraction of entities and relations based on hierarchical sequence labeling," *Acta Scientiarum Naturalium Universitatis Pekinenis*, vol. 57, no. 1, pp. 53–60, 2021.
[54] S. Wu, X. Liang, N. Xie, A. Zhou, and X. Hao, "Annotation method for joint extraction of domain-oriented entity and relation," *J. Comput. Appl.*, vol. 41, pp. 1–9, Oct. 2020.
[55] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 3111–3119.
[56] Y. Gao, Y. Xu, H. Huang, Q. Liu, L. Wei, and L. Liu, "Jointly learning topics in sentence embedding for document summarization," *IEEE Trans. Knowl. Data Eng.*, vol. 32, no. 4, pp. 688–699, Apr. 2020.
[57] S. Lai, "Word and document embeddings based on neural network approaches," 2016, *arXiv:1611.05962*.
[58] Z. Luo, L. Liu, J. Yin, Y. Li, and Z. Wu, "Deep learning of graphs with Ngram convolutional neural networks," *IEEE Trans. Knowl. Data Eng.*, vol. 29, no. 10, pp. 2125–2139, Oct. 2017.
[59] X. Shi, G. Wang, and J. Si, *Mechanical Fault Diagnosis and Typical Case Analysis*. Beijing, China: Chemical Industry Press, 2013.
[60] Z. Huang, *Mechanical Fault Diagnosis Technology and Maintenance Case Selection*. Beijing, China: Chemical Industry Press, 2016.
[61] M. Miwa and M. Bansal, "End-to-end relation extraction using LSTMs on sequences and tree structures," in *Proc. 54th Annu. Meeting Assoc. Comput. Linguistics*, 2016, pp. 1105–1116.
[62] G. Bekoulis, J. Deleu, T. Demeester, and C. Develder, "Adversarial training for multi-context joint entity and relation extraction," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2018, pp. 2830–2836.
[63] W. Huang, X. Cheng, T. Wang, and W. Chu, "Bert-based multi-head selection for joint entity-relation extraction," in *Proc. 8th CCF Int. Conf. Natural Lang. Process. Chin. Comput. (NLPCC)*, 2019, pp. 713–723.



JIANFENG DENG received the master's degree in circuit and systems from the Guangdong University of Technology, Guangdong, China, in 2018, where he is currently pursuing the Ph.D. degree with the Department of Automation. His current research interests include artificial intelligence, deep learning, and natural language processing.

TAO WANG graduated from Sun Yat-sen University, in 2010. He is currently an Associate Professor with the School of Automation, Guangdong University of Technology. His research interests include manufacturing the Internet of Things, industrial big data and knowledge acquisition, intelligent equipment, and robot systems.

ZHUOWEI WANG received the Ph.D. degree in computer system architecture from Wuhan University, Wuhan, China, in 2012. She is currently an Associate Professor with the Institute of Computers, Guangdong University of Technology. Her research interests include high performance computing, low power optimization, and distributed systems.

JIALE ZHOU born in 1996. He is currently pursuing the master's degree with the School of Automation, Guangdong University of Technology. His research interests include natural language processing and robot intelligent fault diagnosis.

LIANGLUN CHENG received the master's degree in automation from the Huazhong University of Science and Technology and the Ph.D. degree in machinery manufacturing and automation from the Changchun Institute of Optical Precision Machinery and Physics, Chinese Academy of Sciences. He is currently a Professor with the Institute of Computer, Guangdong University of Technology. His research interests include the IoT, CPS, and sensor networks.