

## Article

# Knowledge Graph Construction Based on a Joint Model for Equipment Maintenance

Ping Lou , Dan Yu, Xuemei Jiang , Jiwei Hu , Yuhang Zeng  and Chuannian Fan

School of Information Engineering, Wuhan University of Technology, Wuhan 430070, China; louping@whut.edu.cn (P.L.); yudan-264304@whut.edu.cn (D.Y.); huijiwei@whut.edu.cn (J.H.); zengyuhang@whut.edu.cn (Y.Z.); 305838@whut.edu.cn (C.F.)

\* Correspondence: jxm2001@whut.edu.cn

**Abstract:** Under the background of intelligent manufacturing, industrial systems are developing in a more complex and intelligent direction. Equipment maintenance management is facing significant challenges in terms of maintenance workload, system reliability and stability requirements and the overall skill requirements of maintenance personnel. Equipment maintenance management is also developing in the direction of intellectualization. It is important to have a method to construct a domain knowledge graph and to organize and utilize it. As is well known, traditional equipment maintenance is mainly dependent on technicians, and they are required to be very familiar with the maintenance manuals. But it is very difficult to manage and exploit a large quantity of knowledge for technicians in a short time. Hence a method to construct a knowledge graph (KG) for equipment maintenance is proposed to extract knowledge from manuals, and an effective maintenance scheme is obtained with this knowledge graph. Firstly, a joint model based on an enhanced BERT-Bi-LSTM-CRF is put forward to extract knowledge automatically, and a Cosine and Inverse Document Frequency (IDF) based on semantic similarity a presented to eliminate redundancy in the process of the knowledge fusion. Finally, a Decision Support System (DSS) for equipment maintenance is developed and implemented, in which knowledge can be extracted automatically and provide an equipment maintenance scheme according to the requirements. The experimental results show that the joint model used in this paper performs well on Chinese text related to equipment maintenance, with an F1 score of 0.847. The quality of the knowledge graph constructed after eliminating redundancy is also significantly improved.



**Citation:** Lou, P.; Yu, D.; Jiang, X.; Hu, J.; Zeng, Y.; Fan, C. Knowledge Graph Construction Based on a Joint Model for Equipment Maintenance.

*Mathematics* **2023**, *11*, 3748. <https://doi.org/10.3390/math11173748>

Academic Editors: János Sztrik and Francesco Calimeri

Received: 13 June 2023

Revised: 3 August 2023

Accepted: 28 August 2023

Published: 31 August 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

The explosive growth of equipment maintenance data has not only changed the norm for the research and analysis of maintenance but has also brought new opportunities and challenges to the way in which maintenance is carried out [1]. In the field of equipment maintenance, these two requirements place considerable workload and stress on the maintenance engineer. The first requires proper maintenance to optimize equipment performance during the safety period, while the second involves accurate repairs that speed up equipment restoration within a shorter time frame. In practice, this means that the maintainer must regularly verify the operation status of all parts of the device and carry out appropriate servicing to reduce the wear and tear of these components. In addition, the maintenance worker needs to be able to quickly determine the site of the fault and to repair it when the machine appears to be malfunctioning [2]. Obviously, the traditional approach to equipment maintenance still relies heavily on the knowledge and experience of maintainers and specialists in related fields [3–5], which requires them to repeatedly study,

memorize and consult a great quantity of maintenance procedures and to repair manuals in a textual form [6,7]. On the one hand, this is a rather inefficient approach, which not only makes it easy to have inappropriate care and to miss the inspection of certain parts but also makes it difficult to share and transmit the experience of trouble-shooting [8], which leads to poor accuracy and standardization of machine maintenance; on the other hand, a large amount of first-hand knowledge is yet to be managed and utilized effectively, which limits knowledge dissemination and is a waste of data resources. For this reason, there is an urgent demand for a set of strong interpretive Artificial Intelligence technologies to improve the knowledge management of machine maintenance.

As an emerging technology, a knowledge graph (KG) delivers a new scheme to handle equipment maintenance data. The key technologies for building knowledge graphs are Named Entity Recognition (NER) and Relationship Extraction (RE). These technical approaches can be divided into two main categories: pipeline models and joint models. Traditionally, the pipeline approach treats named entity recognition and relationship extraction as a two-step process, first performing the entity recognition and then extracting relationships on the recognized entities. NER aims to identify and classify entities with specific meaning from text, such as the names of people, places, organizations, etc. Rules and manual feature engineering approaches have been the mainstay of early NER research. However, these methods rely on manually defined rules and features, require customization for different languages and domains and are difficult to adapt to large data sets and complex language structures. With the emergence of deep learning, neural network-based approaches are beginning to make significant progress in NER tasks. Among them, Recurrent Neural Network (RNN)-based models, such as Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU), are widely used in NER tasks. These models are able to capture contextual information and to achieve entity recognition by learning feature representations. In addition, with the development of pre-trained language models (PLMs) such as BERT, GPT, etc., the NER task has been further improved. These models can be fine-tuned for downstream NER tasks by being pre-trained on large corpora and by learning rich language representations. RE is concerned with extracting relationships between entities from text. Traditional methods of relation extraction are mainly based on manually designed rules and pattern matching, where relationships are extracted by means of keyword matching, grammar patterns and other means. These methods have the advantage of being highly interpretable, but they require a significant amount of manual work and domain knowledge, and they have difficulty in adapting to complex language structures and diverse types of text. With the growth of machine-learning techniques, machine-learning-based relationship extraction methods have gradually emerged.

However, while the above-mentioned pipeline approach is simple and intuitive, it is susceptible to the propagation of errors and to false matches. To address the problems with the pipeline approach, researchers have proposed the use of joint models. These models treat entity recognition and relation extraction as a unified task and enhance the overall performance through joint optimization. Early methods of entity and relation extraction have relied mainly on rules and pattern matching. These methods depend on manually created rules and patterns, which limit their applicability and ability to be generalized. With the advent of machine learning, researchers have begun to explore the use of machine-learning algorithms to solve the entity relation extraction problem. The first methods used feature-based classifiers, such as Support Vector Machines (SVM) and Maximum Entropy models. These processes extract features from text and train classifiers to predict entity and relationships. In recent years, the rise of deep learning has brought new breakthroughs in knowledge extraction. Researchers started to use neural network models to learn text representations and to perform relation classifications. They achieved high performance by learning feature representations and relation classifiers through end-to-end training. To enhance the performance, researchers started to optimize the objective functions of entity recognition and relation classification together, allowing the two sub-tasks to reinforce each other and improve overall performance. Joint training, shared representation and

sequence annotation are common approaches to joint modelling. In addition, researchers have gradually introduced attention mechanisms to extract key information and the use of pre-trained language models to enhance representation learning.

Additionally, there is a significant amount of redundancy in the extracted entities and relationships. Redundant knowledge takes up a lot of storage space, which can lead to repetition in the computational process and thus reduce efficiency. Additionally, it can produce incorrect and ambiguous knowledge, which significantly affects the quality of the knowledge graph. Therefore, it is necessary to eliminate redundant knowledge before the construction of the knowledge graph. Common techniques include manual review, entity disambiguation and relationship merging. However, manual review is a labor-intensive and time-consuming task. Entity disambiguation and relationship merging methods are widespread. Therefore, methods for resolving entity ambiguity and merging relationships are commonly employed. These can apply predefined rules and heuristic methods to determine whether entities represent the same concept. Alternatively, they can perform similarity-based entity and relationship fusion by calculating the similarity between entities and merging ones with similarity above a threshold.

KGs have the ability to organize and store large amounts of knowledge and can be divided into two categories: general KGs and domain KGs [9], of which general KGs are not limited to specific domains and cover a very wide range of knowledge, while domain-specific KGs are for specific domains such as finance, e-commerce or healthcare [2]. In comparison with general knowledge graphs, domain-specific knowledge graphs can be highly accurate due to their in-depth research and specialized knowledge, as they can capture the details and complexities of specific domains, which can provide more precise results. Consequently, domain-specific knowledge graphs require a larger and more rapidly expanding source of knowledge. Given the diversity of personnel in the industry, with different roles and corresponding operational and business scenarios, domain knowledge graphs need to have a certain level of depth and completeness. Also, they require higher quality standards for knowledge as they are used to support various complex analysis and decision support applications. Moreover, domain knowledge graphs can be presented to users through visualization tools, allowing them to understand and explore domain knowledge more intuitively. They can be used in various scenarios such as knowledge management, intelligent search and recommendation systems.

Currently, there are several challenges in applying KGs in the field of equipment maintenance: on the one hand, equipment maintenance data is large in quantity, knowledge-intensive and involves a series of decision-making activities, most of which are highly dependent on expert knowledge and equipment resources accumulated over a long period of time from working on related projects [1]. This, in turn, results in the diversity of formats and heterogeneity of the relevant knowledge, making it difficult to share and reuse it effectively. As a result, traditional KGs often have to be constructed manually, which is time-consuming and laborious, and this can hardly meet the requirements of an automated KG. On the other hand, the knowledge redundancy problem occurs during automated KG construction, which reduces the KG quality and affects subsequent system [10] development and applications.

Consequently, to address the above issues, the focus of this thesis is to develop a framework for a knowledge graph-based device maintenance system that supports automated construction and the associated design activities. The main contributions of this research can be summarized as follows:

- A maintenance domain-specific word embedding system is trained and a joint extraction model based on BERT-Bi-LSTM-CRF is used to extract entities and relationships.
- A knowledge ontology is designed by exploiting maintenance manuals and analysis reports with predefined relationships and tags to address the problem of ambiguous entity boundaries within this domain.
- An advanced edit distance algorithm has been introduced, which employs Word2vec and a combination of IDF and Cosine for semantic similarity calculations.

- A maintenance system based on a KG using a Browser/Server (B/S) architecture has been developed for equipment maintenance.

The structure of this paper is as follows: Section 2 reviews the relevant work on automatic construction techniques for domain KGs. Section 3 introduces the method of entity and relation joint extraction in the process of constructing domain KGs in Chinese, focusing on the elimination of redundancy in the KG by improving the similarity judgment through the edit distance in the knowledge fusion. Section 4 provides the validation of the application and the extraction effectiveness of the domain KG by developing a prototype system. Section 5 presents the conclusions.

## 2. Related Work

KG is a structured representation method used to describe entities, concepts, relationships and attributes in the real world and to graphically represent their associations, providing a vast knowledge base to support intelligent information processing. It has wide applications in fields such as medicine, agriculture and finance [11]. Knowledge extraction refers to the process of extracting useful information from unstructured text. Named entity recognition (NER) and relation extraction (RE) are core tasks in knowledge extraction. Currently, two main frameworks are widely used: pipeline and joint methods. The pipeline model separates entity recognition and relation extraction into two independent stages. First, entities are identified from the text using an entity recognition model. The relationships between the identified entities are then extracted using a relation extraction model. The advantage of this approach is its simplicity, intuitiveness and ease of implementation and debugging. However, as the two stages are performed independently, errors in the entity recognition stage may affect the subsequent relation extraction. Also, the pipeline model may lose some important contextual information. The joint model treats entity recognition and relation extraction as a unified task, which can exploit the interdependencies between entities and relationships to improve overall performance. It can also improve computational efficiency by sharing features and parameters. This paper reviews the relevant research on the construction and application of KGs, specifically on the knowledge extraction and applications for maintenance KGs.

### 2.1. Knowledge Extraction

NER is a natural language processing technique used to identify entities in text. Existing NER methods mainly include rule-based methods, feature-based methods and neural network-based methods. Early traditional rule-based methods typically required manually designing rules and patterns to match different languages and domains. However, these methods have limitations when dealing with complex language structures and large data sets. In [12], the researchers proposed a CRF-based method, but the model struggles with feature selection, making it difficult to determine which features are most effective for NER performance, as NER involves multiple levels of linguistic information such as parts-of-speech, syntax and semantics. Ritter et al. [13] recognized entities in tweets by manually constructing rules. They experimentally evaluated these rules and, although they had some applications, their manual rule construction required a significant amount of labor and time. In addition, for different languages and domains, rules need to be redesigned, which increases the cost. Furthermore, due to the diverse and complex nature of text, including abbreviations, misspellings, non-standard language and the use of special symbols, rules may not cover all cases, resulting in reduced recognition accuracy. In comparison, neural-network-based methods can automatically learn feature representations from training data, allowing the neural network to extract more valuable features from raw data, thereby enhancing model performance. Moreover, through regularization and other techniques during the training process, neural networks allow for better handling of noise, missing data and unseen samples. Additionally, neural networks can perform end-to-end learning, automatically learning all the steps from the raw input to the final output, which simplifies the model design and reduces the amount of manual tuning and

optimization required. In the field of natural language processing, deep learning started to make its mark in 2013. Researchers started to apply deep-learning models to extract entity and achieved some breakthrough results. Word2Vec was one of the first word vector representation methods [14]. It is based on neural networks and generates word vectors in context. However, it only considers adjacent contexts and has limited ability to distinguish different contexts based on their semantic meaning. Due to this limitation, GloVe [15] was proposed, which learns word vectors by statistically analyzing the global co-occurrence matrix of words. However, it struggles to adapt to different contexts. In 2016, researchers proposed stacking multiple Long Short-Term Memory (LSTM) models to learn hierarchical morphological features, allowing contextual information to be captured. This improved the accuracy of the NER and achieved the best performance on unlabeled corpora at that time [16]. However, the researchers did not consider the contextual relationships of entities, which means that an entity may have different meanings in different contexts. These issues may have an impact on the accuracy of the NER. The Convolutional Neural Networks–Bidirectional Long Short-Term Memory–Conditional Random Fields (CNNs-Bi-LSTM-CRF) network proposed in [17] introduces CNNs to extract local features, which helps to acquire local contextual information and improves the perception of features. It achieved high F1 scores on the CoNLL-2003 English dataset and the Penn Treebank WSJ corpus. But this method employs a fixed-size context window to capture contextual information, which may limit the ability to effectively deal with long-range dependencies. Neither does it consider the representation and use of semantic information, limiting the ability to understand semantics. In [18,19], a deep-learning model that incorporates character-level information has been proposed. The model integrates two different character-level representations extracted from CNN-Bi-LSTM and uses CRF as a decoder to perform sequence labeling tasks by focusing on character-level information. This approach is able to better grasp the internal structure and the features within the words. But it works better for languages such as English and may not give satisfactory results for language-specific tasks such as Chinese word segmentation. The emergence of the new language model BERT [20] has had a significant impact on NER. Using the Transformer model, BERT can simultaneously consider contextual information, thereby allowing semantic and syntactic relationships within sentences to be captured more effectively. In addition, BERT can be pre-trained on large amounts of unlabeled data, and these pre-trained representations can be transferred to various downstream tasks, making it more adaptable to the specific needs of NER in different domains. Souza et al. [21] explored training strategies for the BERT model, combining its transfer capabilities with the structured predictions of the CRF, and achieved good results on the HAREMI dataset. Nevertheless, this method focuses only on Portuguese NER tasks, and its adaptability and effectiveness for Chinese NER tasks have not been validated. Based on the aforementioned literature, it is clear that for entity extraction in the domain of equipment maintenance in Chinese texts, BERT can better represent semantic and syntactic relationships within sentences and has stronger adaptability to specific domains. The Bi-LSTM model can simultaneously take into account contextual information, thus addressing the problem of long-range dependencies more efficiently. And the CRF layer imposes global constraints on the label sequence, considering the dependencies between labels, thereby improving the accuracy of labeling.

RE aims to identify and extract relationships between entities in text. The shortcomings of traditional rule-based methods lie in their difficulty in adapting to complex relationships and the need for manual rule writing. In comparison, using deep neural networks to learn relationships between entities in text has shown major benefits. Early research focused on classification models with simple network structures [22], where researchers used convolutional neural networks (CNNs) to extract relationships. CNNs effectively capture important information in the relation extraction task and can process input data in parallel, speeding up the model's training and inference processes. However, CNNs face challenges in accurately distinguishing between different types of relations when dealing with poly-semantic words, requiring additional processing methods to address the

issue of word ambiguity. Recent methods include distant supervision and deep-learning frameworks. In [23], researchers used remote supervision techniques to automatically label training data, but this introduces noise into the labeling process because the matching of entity pairs to sentences may not be accurate. This may reduce the quality of the training data, and the researchers have not explicitly discussed how to handle poly-semantic words and ambiguous sentences, which may affect the accuracy of the relationship extraction. Later, Huang et al. [24] introduced residual connection blocks after the rolling layers to address the vanishing gradient problem, which improved performance and training efficiency. But this method mainly focuses on sentence-level and entity-level features without fully considering the contextual information of the entities, which may provide a less comprehensive understanding of the relations and is somewhat inadequate when dealing with long-distance relations. It is also limited by the grammar and semantic structure of specific languages. In [25], researchers proposed a pipeline model based on a deep-learning framework. By integrating dependency-tree shortest-path fusion syntax analysis based on LSTM with RE based on NER, they were able to capture long-range dependency relationships and achieve good results in RE. Also, by using the shortest dependency path, this method can focus on the most relevant parts of the sentence, reducing computational and storage costs. On the other hand, this method ignores other important contextual information, which can decrease the classification performance, and it does not mention how to handle poly-semantic words. Zhang et al. [26] considered the positional information of slots in the sentence and combined the sequential LSTM model with an entity position-aware attention mechanism, which is more appropriate for RE. With this combination, the relationship between slots and context can be better captured, leading to a better performance and higher accuracy in slot filling. Obviously, the above pipeline approach of separating NER and RE into two stages ignores the connection between NER and RE characters, which results in the loss of potentially important information and accumulates errors. To avoid these problems, shared models have emerged. By implementing parameter sharing, specifically by sharing the sequential representation of LSTM units in the encoding layer for NER and RE [27], an end-to-end approach has been proposed. This method can extract relationships directly from raw text, and although it does not explicitly discuss how to handle poly-semantic words, it lays the groundwork for subsequent shared models. Zheng et al. [28] proposed a new annotation strategy that transforms the joint learning model involving NER and RE into a full-sequence labeling problem, and achieved good results on regular datasets, providing the basic idea for this paper. An end-to-end neural model was also proposed in [29]. Notably, none of the above studies used transformer-based networks. Recently, researchers have started to fine-tune BERT [30,31], a pre-trained model based on the Transformer architecture. The Transformer is a sophisticated neural network model that can learn contextual information and semantic relationships, thereby improving the accuracy of entity and relationship extraction. It is capable of identifying entities and their relationships simultaneously, avoiding the two independent steps in the pipeline approach. This method has been experimentally validated on several datasets and has shown high performance, demonstrating its effectiveness [32]. Another approach used the Graph Convolutional Network (GCN) and introduced entity attention mechanisms, achieving good performance in end-to-end relationship extraction tasks. However, this method may face challenges in handling long-distance relationships as well as in dealing with ambiguous entity boundaries and noisy data, as the GCN may not accurately capture these details. In addition, the GCN model may not capture domain-specific semantic information, resulting in suboptimal entity and relation extraction performance for certain domains.

## 2.2. Application of Knowledge Graphs

In [33,34], they focus mainly on methods, applications, trends and challenges in extracting information from textual data. The research discusses the applications of information extraction in different domains and the challenges faced. Moreover, it is important for un-

derstanding the latest advances in information extraction and the requirements in different application domains. In certain domains, the application of KGs is widespread. The KG is a large-scale semantic network first proposed by Google in 2012 [35]. In recent years, driven by big data and knowledge acquisition technologies, KGs tailored to specific domains have emerged. For example, in the medical domain, a medical knowledge graph can be constructed to help doctors diagnose and treat diseases. In the financial domain, a financial knowledge graph can be built for risk assessment and investment decisions. In the field of intelligent transport, a transport knowledge graph can be constructed for applications such as traffic management and route planning. Geo-Names is an open global geographic registry covering over 250 countries and more than 10 million geographic locations. The In-Memory Databases (IMDB) [36] integrate the Geographic Names Information System (GNIS) gazetteer with other geographic datasets to create a comprehensive geographic information repository. It provides an intuitive way to browse and explore place names, enabling users to better understand and use this data. Bulla et al. [37] use a span-based joint model and BERT embedding to construct a financial knowledge graph. By exploiting the structure and relationships in the KG, the system can provide accurate and comprehensive answers. Harnoune et al. [38] propose an end-to-end model that uses BERT and CRF layers to extract and analyze knowledge from biomedical clinical notes, thereby constructing a biomedical knowledge graph. The above works construct KGs and demonstrate their benefits in specific industries. The aim of this paper is to develop a prototype system to validate the effectiveness and necessity of building an equipment maintenance KG.

In summary, by analyzing the advantages and disadvantages of the mentioned models and methods, and by considering Chinese textual data from the equipment maintenance details in the paper, it is found that, due to the lack of explicit part-of-speech tagging and unclear entity boundaries in Chinese vocabulary, an ontology for the domain of equipment maintenance needs to be designed. By combining the ontology with predefined relationships and labels, the problem of fuzzy entity boundaries in the equipment maintenance domain is addressed. Furthermore, the BERT model pre-trained on Chinese corpora is selected and fine-tuned to be adapted to the task at hand. An enhanced BERT-Bi-LSTM-CRF joint model is proposed to automatically construct a knowledge graph. Additionally, to improve the quality of the KG, the traditional edit distance algorithm is adapted to compute similarity and to reduce redundancy in knowledge. Finally, a prototype system is designed to validate the approach presented in this thesis.

### 3. Constructing a Knowledge Graph for Equipment Maintenance

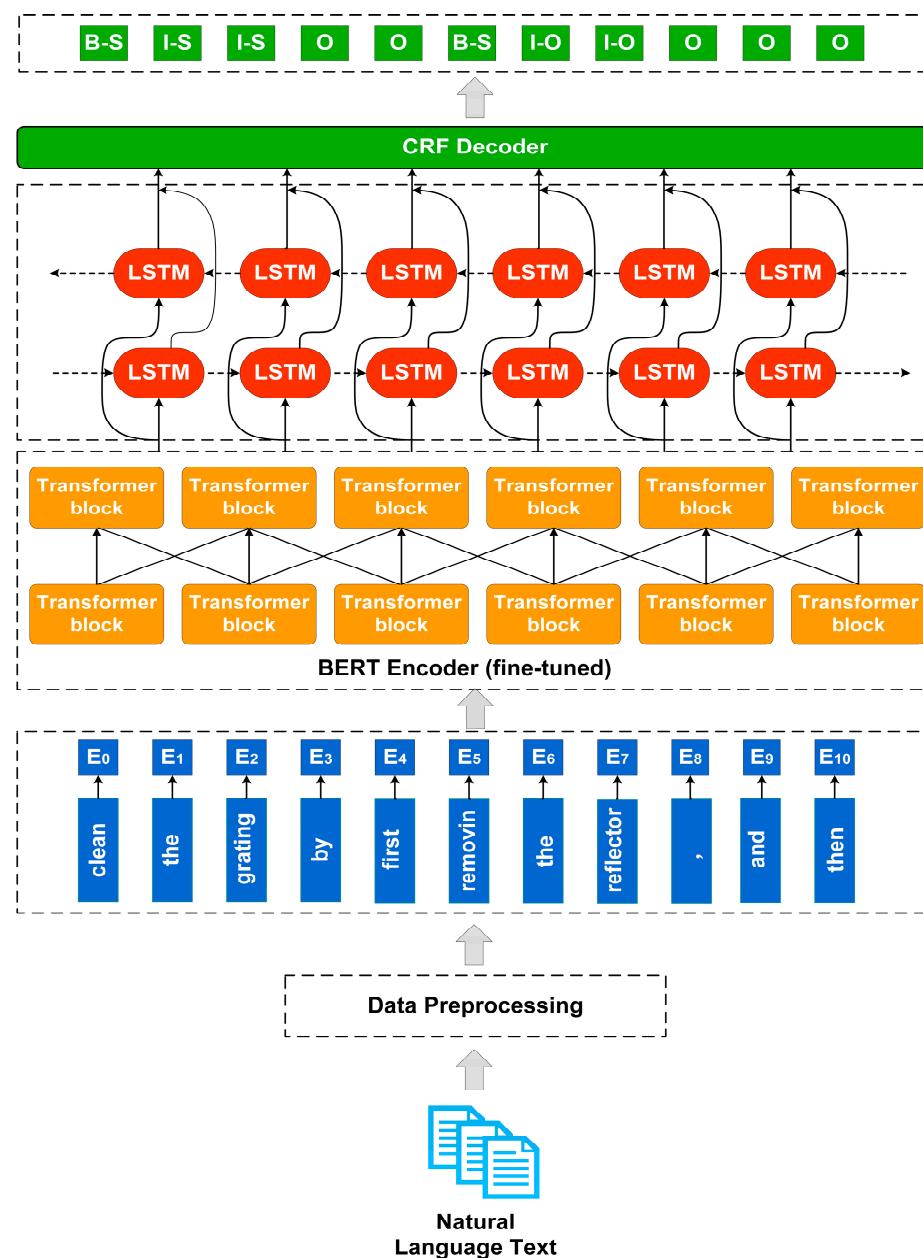
#### 3.1. Knowledge Extraction Based on a Joint Model

The idea of this thesis is to design a knowledge ontology based on the characteristics of equipment maintenance knowledge in order to structure and organize complex domain knowledge and to support the construction of more reliable and accurate KGs in the future. Compared with English, the Chinese character set has a large number of characters and does not have clear delimiters. There is also a lack of explicit part-of-speech tags in Chinese vocabulary, and the boundaries of entities are often unclear. Therefore, in order to solve the problem of fuzzy entity boundaries in the field of equipment maintenance, this paper combines the ontology with predefined relationships and labels to better perform entity and relationship extraction on Chinese texts. Moreover, the choice of the joint model based on BERT-Bi-LSTM-CRF for knowledge extraction is due to the following reasons:

First, considering the high ambiguity and diversity of Chinese texts and the problem of understanding context semantics, we chose the BERT model tailored for Chinese. It has been pre-trained on a large Chinese corpus, making it adaptable to Chinese texts and able to capture the characteristics as well as the patterns of the Chinese language. In addition, BERT, which emphasizes the meaning of words in context as opposed to other models such as Word2vec and Glove, was chosen due to the ambiguity and diversity of natural language, and it has been trained on domain-specific word embedding for equipment maintenance knowledge, potentially improving the quality of subsequent extractions. Second, due to

its bidirectional and synchronous nature, Bi-LSTM further emphasizes the importance of semantics. Finally, in the decoding part, CRF can improve the accuracy of label prediction by learning the rules of NER, in contrast to the traditional Soft-max method, which simply takes the predicted labels with the highest predictor scores.

However, it was found that the pipelined method not only suffered from the cumulative propagation of mistakes, but it also ignored the connection between NER and RE characters and lost potentially valid information, severely affecting the extraction performance. Therefore, to solve the above problems, a joint multi-stage neural network model for knowledge extraction was implemented in the maintenance domain, with several stages of this architecture shown in Figure 1.

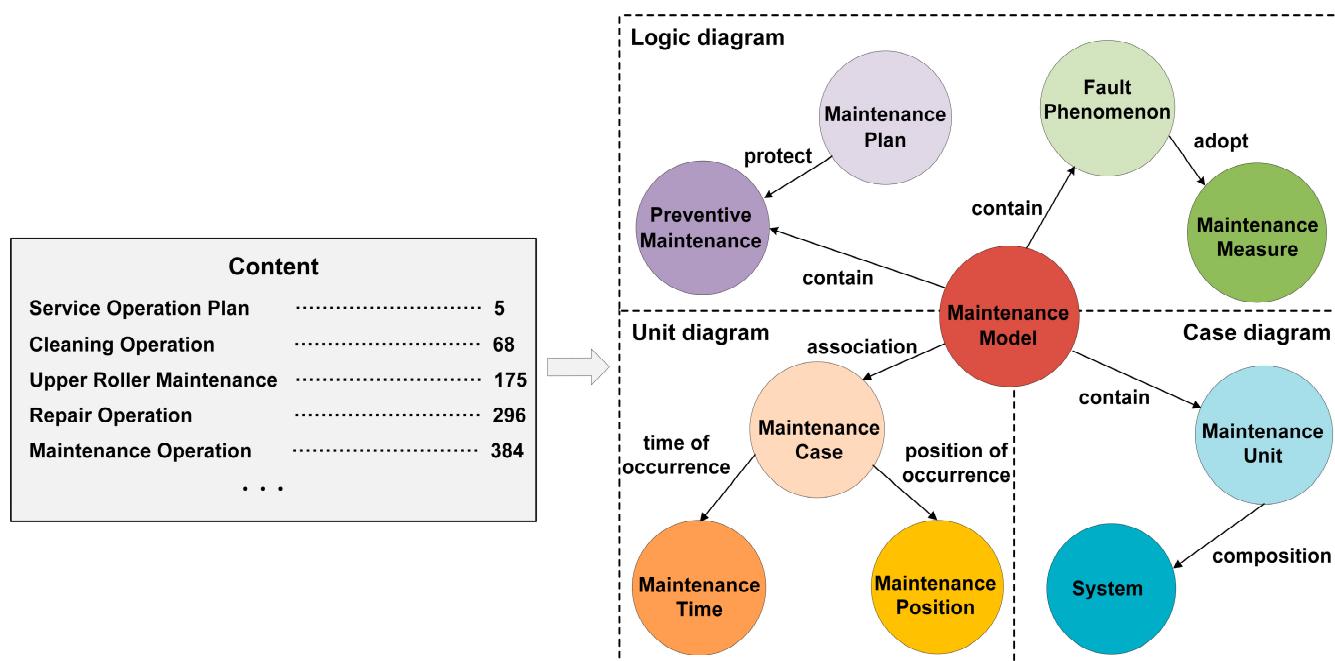


**Figure 1.** Joint extraction model.

### 3.1.1. Text Preprocessing Based on Ontology

Ontology is a conceptualization related to a specific domain, which is an important foundation for building a reliable and sustainable KG that can improve data quality, reduce semantic ambiguity and provide clear and shareable characteristics. This paper focuses on

the design of ontology for maintenance manuals and analysis reports in the maintenance domain, as these documents contain detailed information about the equipment and are rigorously reviewed to ensure accuracy. Moreover, the content is categorized, which helps to clarify the hierarchy of knowledge and promotes knowledge sharing. Therefore, based on the characteristics of the extracted text and through a combination of top-down and bottom-up approaches, we designed a basic model of maintenance knowledge ontology and optimized the model under the guidance of experts and maintenance personnel, as shown in Figure 2. The natural text is complex and irregularly arranged in maintenance manuals and analysis reports. It is difficult to obtain knowledge extraction results satisfactorily if information extraction methods are used directly. Therefore, a set of relationship types was predefined based on ontology to assure its accuracy, which can solve the problem of blurred entity boundaries in Chinese, and the text was annotated according to the ontology on the Doccano platform in order to effectively identify and locate valuable knowledge. An example of partial annotation results is shown in Table 1, which includes entities, entity attributes, relationships between entities, where “contains”, “first step” and “next step” are predefined relationships, and subject and object are entities.



**Figure 2.** Maintenance knowledge ontology design.

**Table 1.** Example of labeling.

| Text   | Predicate  | Object Type | Object                           | Subject Type | Subject                        |
|--|------------|-------------|----------------------------------|--------------|--------------------------------|
| cleaning job needs cleaning grid                                   | contain    | unit        | cleaning the raster              | job          | cleaning work                  |
| cleaning the grating requires removing dust from the grating first | first step | operation   | removing dust from the grating   | unit         | cleaning the grating           |
| then removing dust from the reflector                              | next step  | operation   | removing dust from the reflector | operation    | removing dust from the grating |

### 3.1.2. The BERT Embedding Layer

After the text pre-processing described above, we divided the labeled dataset into a training dataset, a test dataset and an evaluation dataset (for evaluation of the best model). The model was initially refined by trying different segmentation ratios for debugging

purposes, and after comparing their performance, we settled on a ratio of 8:1:1 for the three subsets. In summary, a maintenance corpus with a sufficient amount of data was obtained.

The BERT-based Chinese pre-training model was employed to generate contextually relevant word vectors, which processes text from two directions during training, thus solving the one-way problem of most current word vector generation models and enabling a better understanding of the context and meaning of the text. Moreover, we fine-tuned the BERT model and build a word vector representation layer specifically for the maintenance domain knowledge extraction problem to address the error accumulation and exposure bias problems associated with pipeline models.

First of all, we split the input sequence by adding an identifier (CLS) at the beginning of the sequence and also by using the identifier (SEP) to separate two adjacent clauses. In this case, the output embedding of each word in the sequence consists of three parts: Token Embedding, Segment Embedding and Position Embedding. To jointly extract entities and relations, we added classifier tokens to classify entity and relation, where the relation types are obtained from a set of predefined relations. In addition, we added filters to clean out the non-entity types, as shown in Figure 3. Thus, we obtained a sequence containing both entity labels and relationship labels, and then the sequence vector was fed into the bidirectional Transformer for feature extraction, which yields a sequence vector with rich semantic features. The Transformer is a deep network based on a “self-attentive mechanism”, which mainly adjusts the weight coefficient matrix to capture word characteristics by the degree of association between words in the same sentence:

$$\text{Attention}(Q, K, V) = \text{Softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right)Vj \quad (1)$$

where  $Q$ ,  $K$  and  $V$  are the word vector matrices and  $d_k$  is the embedding dimension.  $\top$  means to transpose; specifically, transpose here means to adjust the elements of row  $i$  and column  $j$  of  $K$  to row  $j$  and column  $i$ , respectively. In contrast to the single-headed attention mechanism, the multi-headed attention mechanism involves the projection of  $Q$ ,  $K$  and  $V$  by several different linear transformations, so that the model can jointly obtain information from different spaces in different locations  $K$ , and finally the different results are stitched together as in formulas (2) and (3):

$$\text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V) \quad (2)$$

$$\text{Multihead}(Q, K, V) = \text{Concat}(\text{head}_1, \dots, \text{head}_n)W^O \quad (3)$$

where  $W_i^Q$ ,  $W_i^K$  and  $W_i^V$  are the three trainable parameter matrices. The input matrix  $X$  is multiplied by  $W_i^Q$ ,  $W_i^K$  and  $W_i^V$  to produce  $Q$ ,  $K$  and  $V$  respectively, which is equivalent to a linear transformation. After obtaining each word vector in the sentence, the sequence of word vector outputs from the BERT layer is fed into the Bi-LSTM module for semantic encoding.

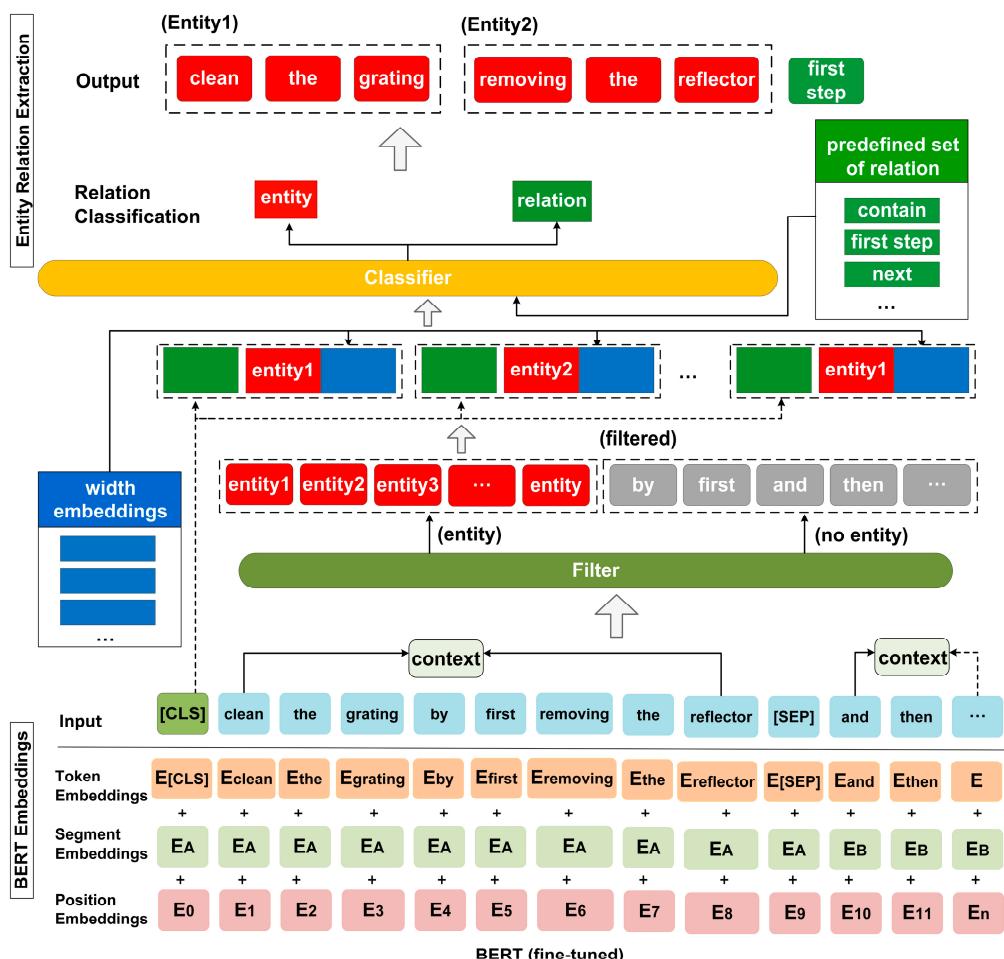
### 3.1.3. The Bi-LSTM Encoding Layer

For the problem of long-range dependency in knowledge extraction, this work employs Bi-LSTM to overcome the obstacle. The basic idea of Bi-LSTM is to take a forward LSTM and a backward LSTM for each sequence of words, and then to merge the results at the same time. In this way, both the forward and the backward information is available at any point in time. It not only solves the problem of exploding and disappearing gradients that occurs when training RNNs, but it also allows for the inability of the one-way LSTM model to process contextual information simultaneously. Assuming that the input sequence is denoted by  $W = \{w_1, w_2, \dots, w_T\}$ , the Bi-LSTM model consists of two separate LSTM layers: one for processing the forward input sequence and another for the backward sequence. The outputs from these two layers are then concatenated across time steps to

generate the final output sequence  $H = \{h_1, h_2, \dots, h_T\}$ . The output of each Bi-LSTM layer can be represented by

$$\text{output} = \left[ \left[ \overrightarrow{h(t)_1}, \overleftarrow{h(t)_n} \right], \dots, \left[ \overrightarrow{h(t)_n}, \overleftarrow{h(t)_1} \right] \right] \quad (4)$$

where  $\overrightarrow{h(t)_i}$  and  $\overleftarrow{h(t)_i}$  denote the output of each hidden layer in the two opposite directions of LSTM,  $n$  is the number of words in the sentence. At this point, we obtain the complete sequence of hidden states, and then we derive the features of the sentence. Bi-LSTM suffers from not handling dependencies between adjacent labels yet. Therefore, the next step is to use a CRF layer for labels to compensate for these drawbacks.



**Figure 3.** Approach towards joint entity and relation extraction through fine-tuned BERT.

### 3.1.4. The CRF Decoding Layer

CRF is a graphical model of joint probability distributions represented by an undirected graph, which can effectively constrain the interactions between prediction labels and model the label sequences. It calculates the probability distribution of the entire sequence and normalizes the local features to the global features to obtain a globally optimal prediction sequence. We assume that  $P$  is the output score matrix of Bi-LSTM, where  $P$  is of size  $n \times k$ ,  $n$  is the number of words,  $k$  is the number of labels, and  $P_{i,j}$  denotes the score of the

corresponding label. We define the sequence of labels  $Y = \{y_1, y_2, \dots, y_n\}$  and use it to score the input sequence  $X = \{x_1, x_2, \dots, x_n\}$  to get the score function as

$$s(X, Y) = \sum_{i=0}^n A_{y_i, y_{i+1}} + \sum_{i=1}^n P_{i, y_i} \quad (5)$$

where  $A$  is the transfer score matrix and  $A_{i,j}$  is the transfer score from label  $i$  to label  $j$ . CRF allows the calculation of the probability between label transformations to avoid an incorrect label order; for example, that the I-place will not be connected to the B-person, and the I-label will not be at the beginning of a name entity. All true label sets are also defined as  $Y$ , and formula 6 is used to calculate the conditional probability of the CRF, where  $y_j \in Y$  denotes the  $j$ -th true label value in  $Y$ . During training, the predicted sequence of the input sequence  $X = \{x_1, x_2, \dots, x_n\}$  is obtained by maximizing the log-likelihood function, then the sequence is decoded by using the Viterbi algorithm, and, finally, the sequence with the highest output prediction score is generated.

$$P(Y|X) = \frac{\exp(s(X, Y))}{\sum_{Y'} \exp(s(X, Y'))} \quad (6)$$

$$Y^* = \arg \max_Y P(Y|X) \quad (7)$$

where  $Y^*$  is the score of the optimal path.

### 3.2. Eliminating Redundancy Based on Knowledge Fusion

From the above procedure, it is not hard to see that knowledge extracted from unstructured equipment maintenance texts may contain a lot of duplicate or similar information. If stored directly in a database, redundant knowledge will increase the size of the KG, thereby reducing the efficiency of querying and increasing the load placed on the system. In addition, redundant knowledge can be a source of unnecessary confusion and misunderstanding, as well as reduce the accuracy of the knowledge graph. If an entity has several different descriptions, these descriptions may be in conflict with each other, with a consequent reduction in the quality of the KG. For instance, given two knowledge triples  $(S_1, P_1, O_1)$  and  $B = (S_2, P_1, O_2)$  in a knowledge graph with the same and similar entities  $S_1, S_2$  or  $O_1, O_2$ , respectively, as the number of knowledge extracted entities grows, the knowledge graph becomes increasingly redundant if similar knowledge triples are not removed. Indeed, in order to effectively fuse and unify this redundant knowledge, knowledge fusion is required when considering the knowledge quality of the KG database. In this paper, we improve the traditional edit distance calculation based on semantic similarity and combine the IDF with the cosine distance to determine the similarity score of entities to derive the final similarity score.

#### Calculation of Similarity Based on Improved Edit Distance

The traditional edit distance cannot determine the degree of similarity at the semantic level, which is particularly important for the semantics of domain KGs, as it can only determine the degree of match of string literal  $d$  by measuring the distance of the string. For this reason, we have introduced Word2vec [34] for unsupervised learning in the traditional editing distance, in which word vectors are trained for specialized domains.

It is commonly desired that the core word match between two similar, but different, entities is correct in their entity similarity matching; in other words, that the core word plays a slightly larger role in the similarity calculation, and that the IDF will evaluate the importance of a word. Thus, the value can be used as a weight to engage in the calculation, using the formula for the computation of word  $idf(w)$  and the sentence vector as follows:

$$idf(w) = \log\left(\frac{D}{D_w + 1}\right) \quad (8)$$

$$\text{vector}(s) = \sum_i^m v(w_i) * \text{idf}(w_i) \quad (9)$$

where  $D$  is the number of sentences in the corpus,  $D_w$  is the number of sentences in which the word occurs,  $v(w_i)$  denotes the vector of  $w_i$  in the sentence and  $\text{idf}(w_i)$  is the value of the  $i$ -th word.

Cosine similarity is assessed by calculating the cosine of the angle between two vectors to assess their similarity. Usually, the more similar the semantics, the higher the cosine score. Assuming that  $v(w_1)$  and  $v(w_2)$  are two n-dimensional vectors calculated by the above formula, i.e.,  $v(w_1) = (x_1, x_2, \dots, x_n)$  and  $v(w_2) = (y_1, y_2, \dots, y_n)$ , the cosine formula is as follows:

$$\cos(\theta) = \frac{\sum_{i=1}^n (x_i * y_i)}{\sqrt{\sum_{i=1}^n (x_i)^2} * \sqrt{\sum_{i=1}^n (y_i)^2}} \quad (10)$$

$$\text{similarity}(X_i, Y_i) = \sum_{i=1}^n \cos(x_i, y_i) \quad (11)$$

where  $\text{similarity}(X_i, Y_i)$  represents the similarity of entities  $X_i$  and  $Y_i$ .

## 4. Prototype System and Analysis

### 4.1. Prototype System

As a means of validating the efficiency of the approach proposed in this paper, we developed a prototype equipment maintenance system based on a KG, which was implemented by adopting a B/S design architecture with integrated Python and Neo4j, enabling easy extension to update the system. The system realizes knowledge extraction, visualization of extraction results, knowledge retrieval and knowledge question and answer. In addition, the system covers the process of data acquisition (real time equipment status data) and storage in a database to support the subsequent establishment of bidirectional data and knowledge-driven maintenance decision. The overall structure is shown in Figure 4.

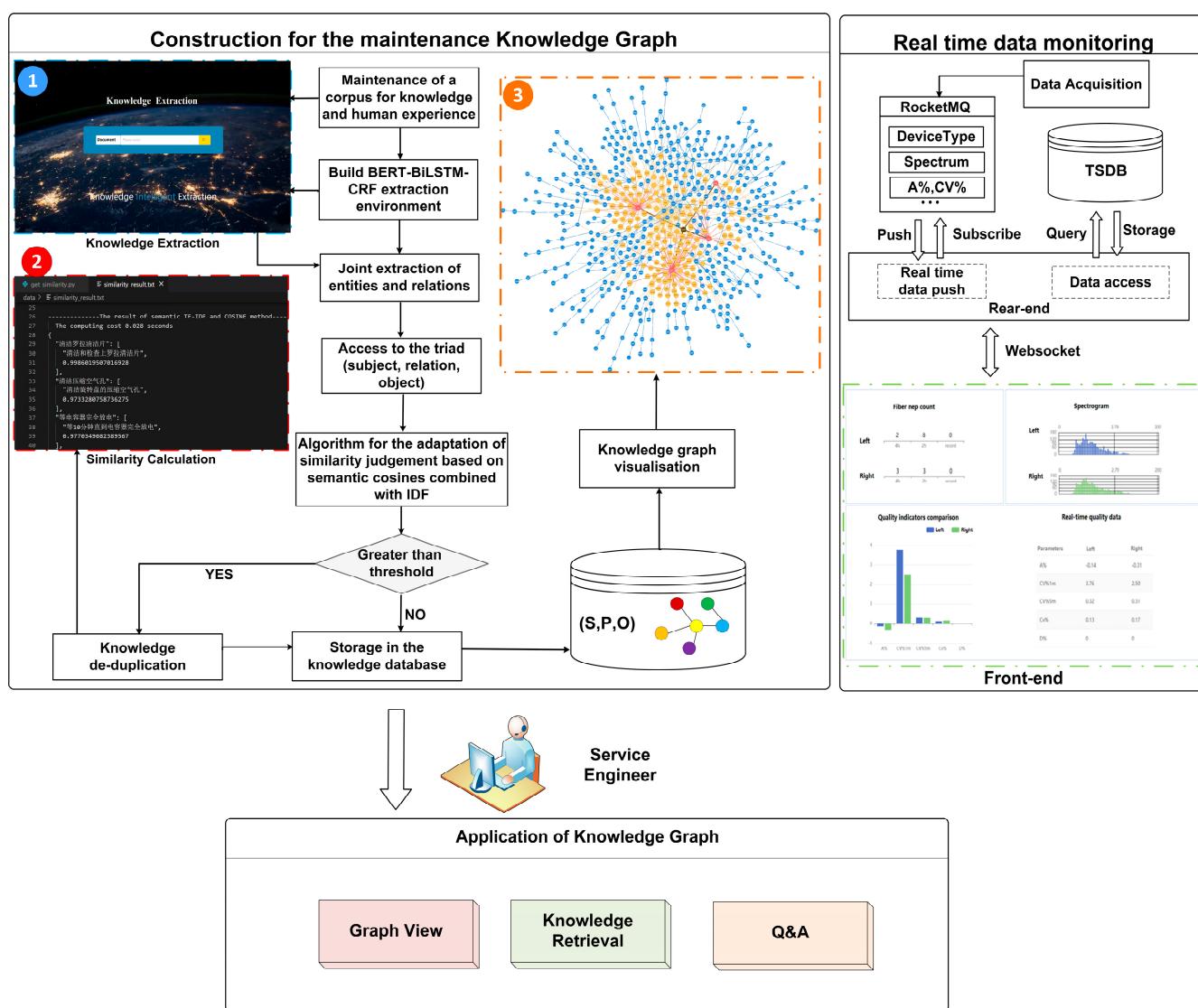
In this work, the equipment maintenance knowledge and data used in the construction of the system are mainly derived from textile machinery industry standard data, maintenance manuals, maintenance analysis reports and maintenance data accumulated by the research team over the years. According to the system of KGs built for equipment maintenance shown in Figure 4, it is clear that the construction of the KG is mainly divided into two aspects. On the one hand, entities and relations are extracted from unstructured text and represented as triples in the environment of the joint model. On the other hand, redundancy reduction is performed in the knowledge fusion phase. The redundant knowledge from the previous extraction is effectively removed by introducing semantics into the traditional edit distance algorithm as well as by combining the IDF and Cosine to compute similarity, which is finally stored in the KG. Following the above steps, the quality of the KG can be upgraded, thus providing a foundation for reasoning about applying the graph in downstream tasks.

Naturally, experiments were carried out to evaluate the performance of the prototype system by investigating (1) the quality of the constructed KG of equipment maintenance and (2) the application of KG in the maintenance field.

### 4.2. Quality Analysis of Knowledge Extraction

In this paper, the joint model was used for knowledge extraction for Chinese text in the maintenance domain. To verify the efficiency of the proposed method, appropriate parameter adjustments were made in this paper according to the training of the joint model, and the learning rate was set  $2 \times 10^{-6}$  to better stabilize and control the training process of the model after multiple trainings. Moreover, the Adam optimizer was selected, which

has better performance and convergence speed in the training of the neural network and is especially suitable for the training of such large data and complex models as Chinese text, helping to speed up the convergence and jump out of the local optimal solution. The detailed parameter settings are shown in Table 2. Next, in this work, the Chinese corpus was divided into training, test and validation sets in the ratio of 8:1:1. Then, the joint model based on BERT-Bi-LSTM-CRF was trained on the same dataset with the pipeline model [39], the CNN-Bi-LSTM-CRF model and the Bi-LSTM-CRF model with the elimination of the BERT module, and the experimental results are shown in Table 3. From the experimental results, the Accuracy, Recall and F1 scores of the joint model are higher than those of the other three models, indicating that the joint model achieves better extraction performance for Chinese text in the domain of maintenance. The reasons are analyzed as follows:



**Figure 4.** Process and application of building a KG. ① is for knowledge extraction; ② is to calculate the similarity of the entities; ③ is the knowledge graph visualization.

The pipeline model has drawbacks, including a high dependency on the previous trained model, which can cause errors to accumulate, and it is hard to train end-to-end. Also, each component needs to be tuned and optimized separately, thus increasing the complexity. In this paper, the joint model based on the BERT Chinese pre-trained model was employed, as it is an end-to-end model that can be trained directly from raw text to

label sequences. Compared with the pipeline model, it requires no manual construction and processing of intermediate results, which decreases the risk of error transmission. Meanwhile, end-to-end training can better exploit the overall capability of the model and increase its performance. This approach fully considers the influence of contextual information on entities, which is beneficial for extracting features from Chinese text. The BERT model used in this paper also includes a classifier for relations and entities as well as a filter for non-entities to improve the predictive ability of the model for entities and relations. Furthermore, to reduce the burden of manual analysis, improve the reliability and accuracy of knowledge extraction and to facilitate the automatic identification of specific types of entities and the extraction of predefined relationships, a set of relationships was predefined and integrated into the contextual representation, effectively avoiding the problem of entity boundary ambiguity in the domain of equipment maintenance. This information was used as input for Bi-LSTM to generate predictions, and the final entity and relationship information was obtained by extracting the globally optimal label sequence of the target sentence from the CRF layer.

**Table 2.** Description of the relevant parameters in the experiment.

| Parameter Name          | Parameter Value    |
|-------------------------|--------------------|
| Batch size              | 32                 |
| Gradient                | 0.5                |
| Inactivation rate       | 0.5                |
| Learning rate           | $2 \times 10^{-6}$ |
| Maximum sequence length | 128                |
| Length of LSTM          | 128                |
| Optimizer algorithm     | Adam               |

**Table 3.** Comparison of the effectiveness with the pipeline model.

| Model                           | Precision% | Recall% | F1%    |
|---------------------------------|------------|---------|--------|
| BERT-Bi-LSTM-CRF joint model    | 96.15%     | 75.76%  | 84.75% |
| BERT-Bi-LSTM-CRF pipeline model | 80.22%     | 64.95%  | 71.18% |
| CNN-Bi-LSTM-CRF                 | 73.12%     | 74.36%  | 73.73% |
| Bi-LSTM-CRF                     | 69.25%     | 61.93%  | 65.47% |

On the other hand, for Chinese texts in the domain of equipment maintenance, the model employed in this paper outperforms the CNN-Bi-LSTM-CRF model. The reason is that the CNN model can only capture contextual information through local features and cannot fully understand the complexity of the Chinese language. Not only does the CNN module use fixed window sizes for convolutional kernels when processing text, which restricts its ability to flexibly learn representations of words of different lengths, it also has poorer ability to handle out-of-vocabulary words and spelling errors due to the use of static word vector representations. In contrast, the joint model used in this paper can carry out global context modeling on input sequences and utilizes character-level WordPiece embedding. Compared with the CNN-Bi-LSTM-CRF model, it has substantial benefits in terms of contextual understanding, word vector representation and feature extraction capabilities.

Compared with the Bi-LSTM-CRF model, the BERT module used in this paper can learn broader semantic word representations, thereby improving the performance in extraction tasks. BERT, based on the Transformer architecture, can consider the context information both before and after a word simultaneously, resulting in a better understanding of the relationships between entities. The ability to understand context allows the joint model to better capture the semantic and contextual information in sentences, thereby raising the accuracy of extraction. As can be seen from Table 3, the joint model used in this paper achieves much higher accuracy in extraction than the Bi-LSTM-CRF model.

#### 4.3. Assessment of the Entity Similarity Algorithm

In the process of building a knowledge graph, there are often many duplicate or similar pieces of information in the triple data obtained from knowledge extraction. If these redundant pieces of knowledge are stored directly in the database, they can cause unnecessary confusion and misunderstanding. Moreover, if an entity has several different descriptions, the knowledge graph may generate ambiguous or contradictory knowledge, which can greatly reduce its accuracy and trustworthiness. Therefore, using the triple data generated by the aforementioned knowledge extraction process as an example, this paper proposes the use of semantics combined with the IDF and Cosine to compute entity similarity.

According to the experimental results shown in Table 4, it is clear that the method employed in this paper performs better than that of Jaccard and Cosine [40] in determining the similarity between entities. For similar entities, the method proposed in this work gives higher values of similarity, while for confusing entities, this method is able to distinguish between them in a better way. The reason for this is that the method weights each word in the text, thereby converting different words into vector representations with similar semantics, thus avoiding errors in similarity calculations caused by different expressions and overcoming the problem of lexical diversity. In addition, this method also decreases the effect of differences in text length. Following several sets of extensive experimental data, the threshold  $w$  is set to 0.8 in this study. If the calculation is greater than  $w$ , the entity is considered redundant and must be de-duplicated. To demonstrate the importance and effectiveness of the algorithm, a statistical analysis was performed comparing the total number of nodes in the KG and the number of redundant nodes with and without the implementation of the three redundancy removal methods mentioned above. The results are presented in Table 5. It is evident that the knowledge graph without any redundancy removal has the highest ratio of redundant nodes to the total nodes, indicating the highest degree of redundancy. This significantly reduces the quality and credibility of the knowledge graph, as well as increasing the storage space and query time, affecting the performance and response speed of the subsequent system. In comparison, the proposed algorithm shows the most substantial reduction in redundancy compared with the other two methods, which greatly improves the quality of the KG and increases the efficiency of storage and querying.

**Table 4.** Example of entity similarity computation.

| Entity <sub>1</sub>                                | Entity <sub>2</sub>                                  | Sim <sub>Cosine+IDF</sub> | Sim <sub>Cosine</sub> | Sim <sub>Jaccard</sub> |
|--|--|---------------------------|-----------------------|------------------------|
| Clean roller cleaning sheet                        | Clean and inspect cleaning sheet of the upper roller | 0.9986                    | 0.8512                | 0.5756                 |
| Wait for the capacitor to be completely discharged | Wait 10 min until the capacitor has been discharged  | 0.9770                    | 0.7494                | 0.5384                 |
| Cleaning of compressed air hole                    | Clean the compressed air holes of the rotating disc  | 0.9733                    | 0.7074                | 0.6364                 |
| Unlock the padlock on the removal master switch    | Unlock and remove padlock                            | 0.8148                    | 0.7498                | 0.6363                 |

**Table 5.** Redundancy of Knowledge Graph Nodes.

| De-Redundancy Methods | Redundant Nodes (RN) | Total Nodes (TN) | RN/TN  |
|-----------------------|----------------------|------------------|--------|
| No                    | 1762                 | 2965             | 59.43% |
| Jaccard               | 1026                 | 2229             | 46.03% |
| Cosine                | 284                  | 1487             | 19.10% |
| Cosine with IDF       | 98                   | 1301             | 7.53%  |

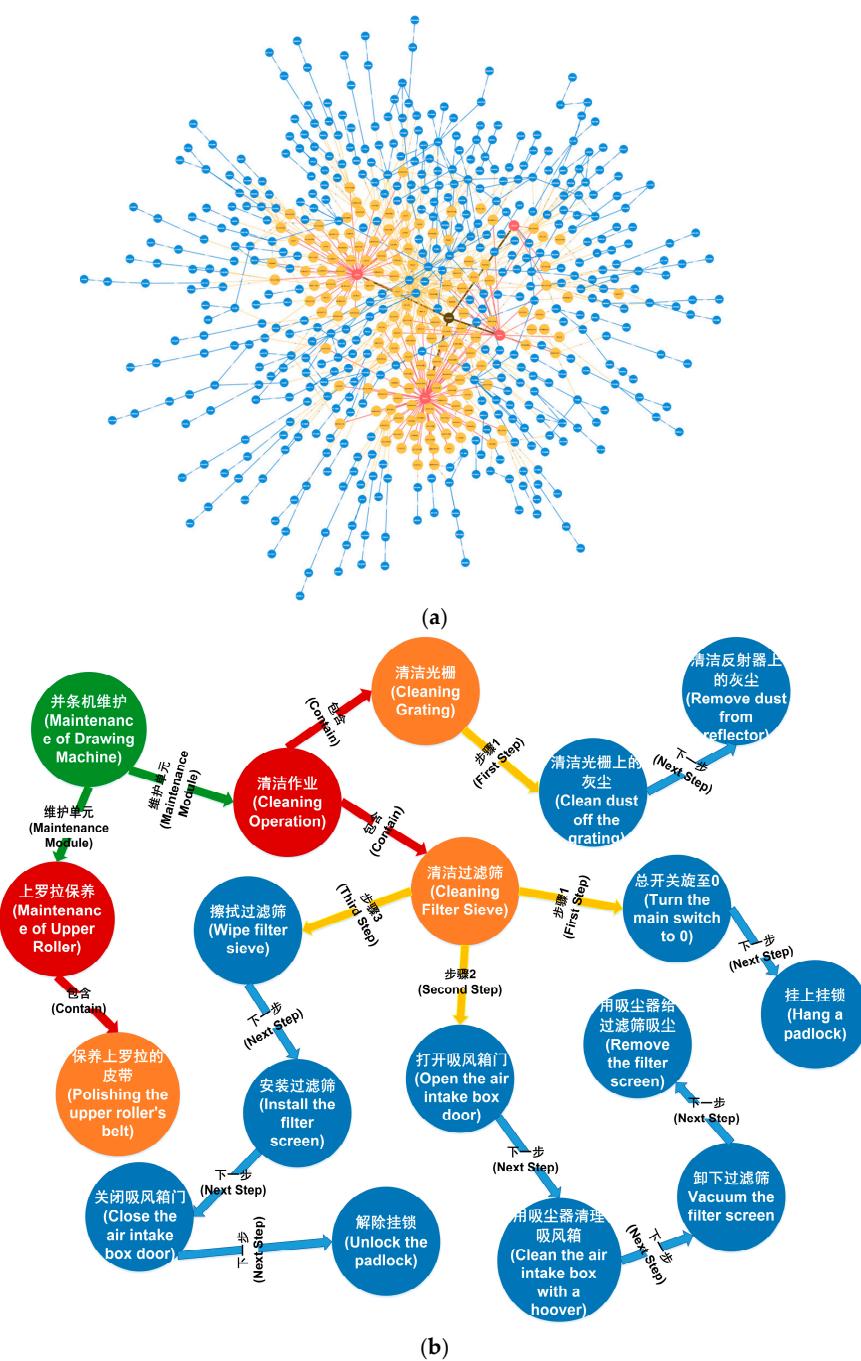
#### 4.4. Case Study

In this article, an equipment maintenance system is set up to provide a detailed demonstration of the construction and application of the KG in the maintenance process, as shown in Figure 4. In Figure 4, ① represents the page used for knowledge extraction, where maintenance engineers can upload texts related to equipment maintenance, such as maintenance manuals, maintenance reports or textual descriptions of their practical experience. The uploaded texts are processed by the BERT-Bi-LSTM-CRF joint model to extract entities and relationships, which are then represented as triplets. Each triplet consists of a head entity, a tail entity and the relationship between them. At this stage, ambiguous or conflicting knowledge may be found in the extracted triplets due to high redundancy. Not only does this take up a large amount of storage space, but it also has a major impact on the quality and timeliness of the subsequent construction of the KG. Hence, to calculate the similarity between entities, the semantic-based Cosine similarity algorithm is used in combination with the IDF, which is shown in ②. If the calculated similarity exceeds a predefined threshold, it is determined that these entities are redundant and need to be de-duplicated, i.e., only one entity is retained while the rest of the similar entities are discarded. This process eliminates the effects of redundant knowledge and enables the construction of a concise and accurate KG. Once the above steps have been completed, the de-duplicated triplets are stored in the database (③). At this point, the equipment maintenance KG is constructed and can be accessed by the system to implement maintenance functionalities.

This paper takes the maintenance manual and maintenance report of the warping machine as an example. It contains operations such as top roller maintenance, overhaul, cleaning and repair. Entities and relationships are extracted using a joint model, and the similarity between entities is computed for de-duplication. The data is then stored in a graph database. Neo4j was chosen for this project because it can better represent the associations and attributes between entities. It also offers good flexibility and scalability, making it suitable for dealing with the characteristics of the constantly growing and changing textual data in equipment maintenance. In addition, Neo4j provides support for Atomicity, Consistency, Isolation and Durability (ACID) transactions. This means that Neo4j can ensure data integrity and consistency as data is modified, which is critical for a reliable and consistent asset maintenance system. The overall graph representation shown in Figure 5a,b specifically lists the extracted entities and relationships (the objects extracted in this paper are Chinese texts, partially translated into English for a more intuitive understanding). It can be seen that each node represents an entity and that the relationships between nodes are from subject to object. For example, top roller maintenance and cleaning operations are two aspects of warping machine maintenance, while cleaning operations include more specific units such as cleaning, grating and cleaning the filter sieve. Each unit has specific operating procedures. Specifically, the original text “Cleaning grating requires first clean dust off the grating and then remove dust from reflector” is extracted by the joint model. “Cleaning grating”, “Clean dust off grating” and “Remove dust from reflector” are entities, while “first step” and “next step” are predefined relationships between different entities. It is obvious that the “first step” of “Cleaning grating” is “Clean dust off grating” and the “next step” of “Clean dust off grating” is “Remove dust from reflector”.

In summary, the construction of the KG for maintaining the warping machine is complete, and the system is ready for use. To provide a comfortable interactive experience for service engineers, the system offers a convenient knowledge search, avoiding the drawbacks of poor readability and the lack of flexibility in Cypher. Maintenance personnel can quickly find the information they need, improving work efficiency. The system also provides a basic knowledge question-and-answer function, allowing maintainers to ask questions about maintenance-related knowledge of the machine and providing guidance and support. There is a description of the system based on specific examples. In Figure 6a, the graph shows all the cleaning units involved in the cleaning operation, and the content of each unit is displayed in the “Detailed Information” panel, making it convenient for

maintainers to view. Figure 6b shows the graph of the cleaning of the exhaust fan, and the entities and relationships in the graph are transformed into natural language and displayed in the “Detailed Information” panel, making it easier to read. Here are the specific steps for cleaning the exhaust fan. Figure 6c shows the question-and-answer page, where enquiries can be made about the entities and relationships in the KG, such as “What does “cleaning operation” involve?” and “How do you clean the exhaust fan?” Furthermore, the system has added an important real-time data monitoring function, which monitors key parameters such as CV, AV and spectrum. It provides real-time feedback on the warping machine’s operating status, providing reliable data collection for future bidirectional knowledge and data driving.



**Figure 5.** Knowledge graph visualization: (a) Knowledge graph overview; (b) Zoomed-in view of the knowledge graph.

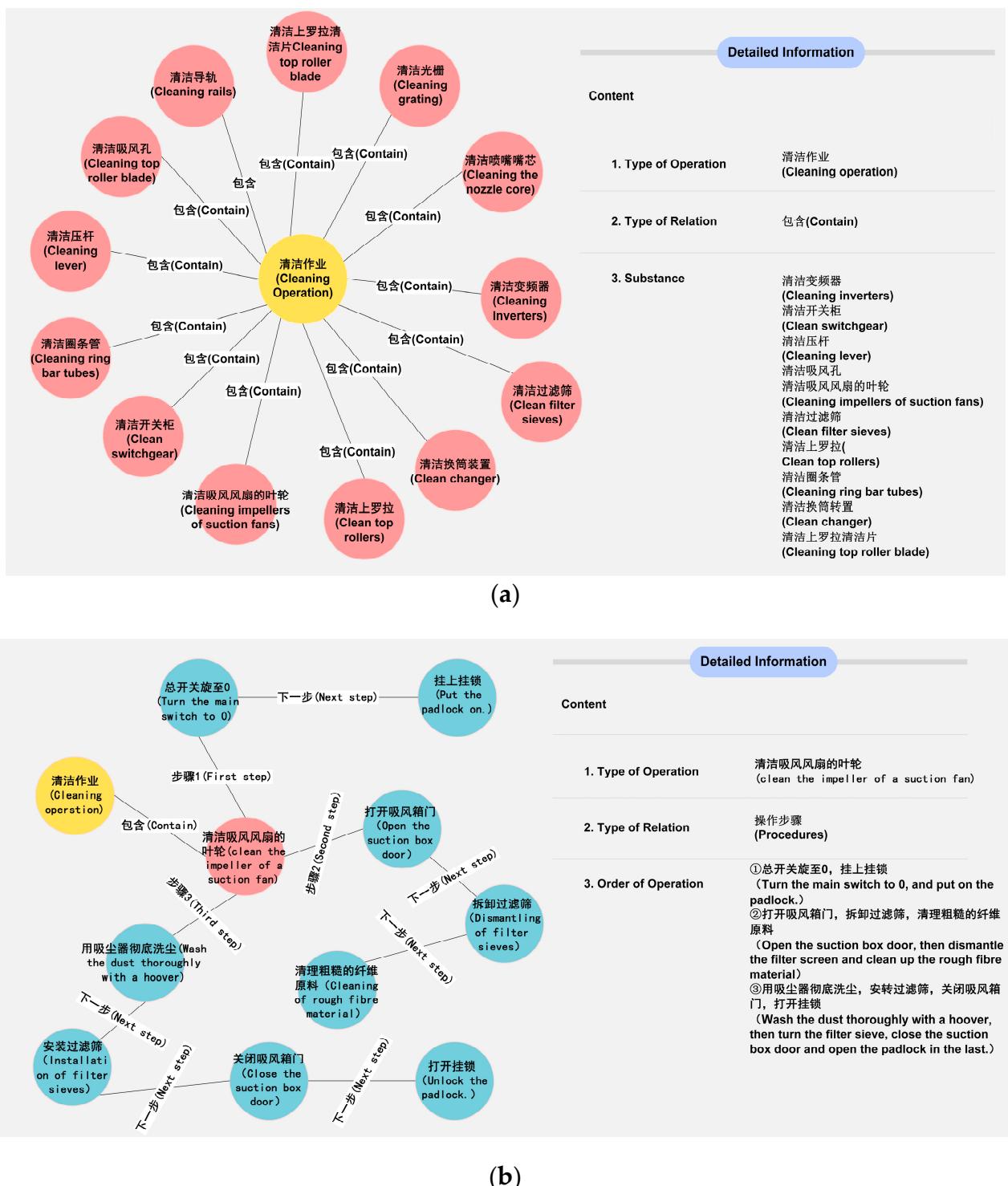


Figure 6. Cont.



(c)



(d)

**Figure 6.** Knowledge graph visualization: (a) Cleaning units in the cleaning operation; (b) Cleaning steps of cleaning the drum-changing device; (c) Question and answer; (d) Real-time data monitoring.

## 5. Conclusions

Current equipment maintenance methods rely heavily on the knowledge and experience of maintenance personnel and experts in related fields. They must repeatedly

study, memorize and refer to a number of text-based maintenance manuals and analysis reports. Not only is this inefficient, but it also makes it difficult to share and transfer troubleshooting experience. Compared with the above discussed pipeline model based on BERT-Bi-LSTM-CRF, the CNN-Bi-LSTM-CRF model and the Bi-LSTM-CRF model without the BERT module, the joint model in this paper achieves the highest accuracy, recall and F1 scores. This indicates that the joint model performs well for Chinese texts in the maintenance domain and eliminates redundancy to build a high-quality KG, thereby improving knowledge management in equipment maintenance and greatly enhancing the sharing of equipment maintenance knowledge and the efficiency. Previous studies have often relied on expert interviews or manual methods to construct KGs, especially in the face of global events such as COVID-19, where traditional face-to-face activities may undergo significant changes.

Consequently, the focus of this paper was to develop an automatic framework for constructing a Chinese KG specifically for equipment maintenance. This framework effectively manages knowledge and provides a case to demonstrate how it can assist service engineers in equipment maintenance. In summary, the approach has the following advantages:

(1) Domain-specific word embedding is trained for the maintenance function, and a joint extraction model based on BERT-Bi-LSTM-CRF is used to extract entities and relationships. This architecture improves the performance of knowledge extraction by avoiding the accumulation of errors caused by pipeline models, which allows the capture and representation of knowledge in the maintenance domain. In addition, a knowledge ontology is designed based on maintenance manuals and analysis reports. Combining these with the ontology-added tags and predefined relationships to the knowledge, this solves the problem of ambiguous entity boundaries in this domain.

(2) An enhanced edit distance algorithm that introduces semantics and combines the IDF and Cosine is proposed to eliminate knowledge redundancy and promote the quality of the KG.

(3) A system adopts a B/S design architecture, which can be flexibly extended to update the system, and includes convenient knowledge retrieval, detailed information display, knowledge question and answer and real-time data monitoring functions to facilitate the interaction of maintenance engineers.

In future work, we will consider analyzing the real-time data collected and the signs from the equipment to establish links and to support decision making based on both knowledge and data.

**Author Contributions:** Conceptualization, P.L. and D.Y.; methodology, P.L., D.Y. and J.H.; validation, P.L., D.Y. and X.J.; formal analysis, C.F., J.H. and Y.Z.; investigation, P.L., D.Y. and X.J.; writing—original draft preparation, P.L. and D.Y.; writing—review and editing, X.J., Y.Z. and J.H.; visualization, D.Y. and C.F.; supervision, P.L. and J.H.; project administration, P.L. and J.H.; funding acquisition, P.L. and J.H. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the National Natural Science Foundation Committee (NSFC) of China, grant number 52075404, and the National Key Research and Development Project of China, grant number 2020YFB1710804.

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Girodon, J.; Monticolo, D.; Bonjour, E.; Perrier, M. An Organizational Approach to Designing an Intelligent Knowledge-Based System: Application to the Decision-Making Process in Design Projects. *Adv. Eng. Inf.* **2015**, *29*, 696–713. [[CrossRef](#)]
2. Lin, Q.; Zhang, Y.; Yang, S.; Ma, S.; Zhang, T.; Xiao, Q. A Self-Learning and Self-Optimizing Framework for the Fault Diagnosis Knowledge Base in a Workshop. *Robot. Comput.-Integr. Manuf.* **2020**, *65*, 101975. [[CrossRef](#)]
3. Chen, H.; Luo, X. An Automatic Literature Knowledge Graph and Reasoning Network Modeling Framework Based on Ontology and Natural Language Processing. *Adv. Eng. Inf.* **2019**, *42*, 100959. [[CrossRef](#)]

4. Robinson, M.A. How Design Engineers Spend Their Time: Job Content and Task Satisfaction. *Design Studies* **2012**, *33*, 391–425. [[CrossRef](#)]
5. Guo, L.; Yan, F.; Li, T.; Yang, T.; Lu, Y. An Automatic Method for Constructing Machining Process Knowledge Base from Knowledge Graph. *Robot. Comput.-Integr. Manuf.* **2022**, *73*, 102222. [[CrossRef](#)]
6. Tennant, J.P.; Crane, H.; Crick, T.; Davila, J.; Enkhbayar, A.; Havemann, J.; Kramer, B.; Martin, R.; Masuzzo, P.; Nobes, A.; et al. Ten Hot Topics around Scholarly Publishing. *Publications* **2019**, *7*, 34. [[CrossRef](#)]
7. Yang, X.; Zhao, S.; Cheng, B.; Wang, X.; Ao, J.; Li, Z.; Cao, Z. A General Solution and Practice for Automatically Constructing Domain Knowledge Graph. In Proceedings of the 2020 IEEE 6th International Conference on Computer and Communications (ICCC), Chengdu, China, 11–14 December 2020; pp. 1675–1681.
8. Zhu, X.; Li, Z.; Wang, X.; Jiang, X.; Sun, P.; Wang, X.; Xiao, Y.; Yuan, N.J. Multi-Modal Knowledge Graph Construction and Application: A Survey. *IEEE Trans. Knowl. Data Eng.* **2022**, *1*–20. [[CrossRef](#)]
9. Jiang, P.; Ruan, X.; Feng, Z.; Jiang, Y.; Xiong, B. Research on Online Collaborative Problem-Solving in the Last 10 Years: Current Status, Hotspots, and Outlook—A Knowledge Graph Analysis Based on CiteSpace. *Mathematics* **2023**, *11*, 2353. [[CrossRef](#)]
10. Chen, Y.; Ge, X.; Yang, S.; Hu, L.; Li, J.; Zhang, J. A Survey on Multimodal Knowledge Graphs: Construction, Completion and Applications. *Mathematics* **2023**, *11*, 1815. [[CrossRef](#)]
11. Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.S.; Dean, J. Distributed Representations of Words and Phrases and Their Compositionality. In Proceedings of the Advances in Neural Information Processing Systems, Lake Tahoe, NV, USA, 5–8 December 2013; Volume 26.
12. Ratinov, L.; Roth, D. Design Challenges and Misconceptions in Named Entity Recognition. In Proceedings of the Thirteenth Conference on Computational Natural Language Learning (CoNLL-2009), Boulder, CO, USA, 4–5 June 2009; pp. 147–155.
13. Ritter, A.; Clark, S.; Mausam; Etzioni, O. Named Entity Recognition in Tweets: An Experimental Study. In Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, Scotland, UK, 27–31 July 2011; pp. 1524–1534.
14. Ramamurthy, R.; Lübbing, M.; Bell, T.; Gebauer, M.; Ulusay, B.; Uedelhoven, D.; Khameneh, T.D.; Loitz, R.; Pielka, M.; Bauckhage, C.; et al. Automatic Indexing of Financial Documents via Information Extraction. In Proceedings of the 2021 IEEE Symposium Series on Computational Intelligence (SSCI), Orlando, FL, USA, 5–7 December 2021; pp. 1–5.
15. Pennington, J.; Socher, R.; Manning, C. GloVe: Global Vectors for Word Representation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, 25–29 October 2014; pp. 1532–1543.
16. Lample, G.; Ballesteros, M.; Subramanian, S.; Kawakami, K.; Dyer, C. Neural Architectures for Named Entity Recognition. *arXiv* **2016**, arXiv:1603.01360.
17. Ma, X.; Hovy, E. End-to-End Sequence Labeling via Bi-Directional LSTM-CNNs-CRF. *arXiv* **2016**, arXiv:1603.01354.
18. Rei, M.; Crichton, G.K.O.; Pyysalo, S. Attending to Characters in Neural Sequence Labeling Models. *arXiv* **2016**, arXiv:1611.04361.
19. Cho, M.; Ha, J.; Park, C.; Park, S. Combinatorial Feature Embedding Based on CNN and LSTM for Biomedical Named Entity Recognition. *J. Biomed. Inf.* **2020**, *103*, 103381. [[CrossRef](#)] [[PubMed](#)]
20. Devlin, J.; Chang, M.-W.; Lee, K.; Toutanova, K. BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding. *arXiv* **2019**, arXiv:1810.04805.
21. Souza, F.; Nogueira, R.; Lotufo, R. Portuguese Named Entity Recognition Using BERT-CRF. *arXiv* **2020**, arXiv:1909.10649.
22. Liu, C.; Sun, W.; Chao, W.; Che, W. Convolution Neural Network for Relation Extraction. In Proceedings of the Advanced Data Mining and Applications, Hangzhou, China, 14–16 December 2013; Motoda, H., Wu, Z., Cao, L., Zaiane, O., Yao, M., Wang, W., Eds.; Springer: Berlin/Heidelberg, Germany, 2013; pp. 231–242.
23. Zeng, D.; Liu, K.; Chen, Y.; Zhao, J. Distant Supervision for Relation Extraction via Piecewise Convolutional Neural Networks. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, Lisbon, Portugal, 17–21 September 2015; pp. 1753–1762.
24. Huang, Y.Y.; Wang, W.Y. Deep Residual Learning for Weakly-Supervised Relation Extraction. *arXiv* **2017**, arXiv:1707.08866.
25. Yan, X.; Mou, L.; Li, G.; Chen, Y.; Peng, H.; Jin, Z. Classifying Relations via Long Short Term Memory Networks along Shortest Dependency Path. *arXiv* **2015**, arXiv:1508.03720.
26. Zhang, Y.; Zhong, V.; Chen, D.; Angeli, G.; Manning, C.D. Position-Aware Attention and Supervised Data Improve Slot Filling. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), Copenhagen, Denmark, 9–11 September 2017; pp. 35–45.
27. Miwa, M.; Bansal, M. End-to-End Relation Extraction Using LSTMs on Sequences and Tree Structures. *arXiv* **2016**, arXiv:1601.00770.
28. Zheng, S.; Wang, F.; Bao, H.; Hao, Y.; Zhou, P.; Xu, B. Joint Extraction of Entities and Relations Based on a Novel Tagging Scheme. *arXiv* **2017**, arXiv:1706.05075.
29. Zeng, X.; Zeng, D.; He, S.; Liu, K.; Zhao, J. Extracting Relational Facts by an End-to-End Neural Model with Copy Mechanism. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Melbourne, Australia, 15–20 July 2018; pp. 506–514.
30. Li, X.; Yin, F.; Sun, Z.; Li, X.; Yuan, A.; Chai, D.; Zhou, M.; Li, J. Entity-Relation Extraction as Multi-Turn Question Answering. *arXiv* **2019**, arXiv:1905.05529.
31. Eberts, M.; Ulges, A. Span-Based Joint Entity and Relation Extraction with Transformer Pre-Training. *arXiv* **2019**, arXiv:1909.07755.

32. Wang, Q.; Lv, L.; Yu, B.; Li, S. End-to-End Relation Extraction Using Graph Convolutional Network with a Novel Entity Attention. In Proceedings of the 2020 IEEE 6th International Conference on Computer and Communications (ICCC), Chengdu, China, 11–14 December 2020.
33. Zhao, H.; Pan, Y.; Yang, F. Research on Information Extraction of Technical Documents and Construction of Domain Knowledge Graph. *IEEE Access* **2020**, *8*, 168087–168098. [[CrossRef](#)]
34. Abdullah, M.H.A.; Aziz, N.; Abdulkadir, S.J.; Alhussian, H.S.A.; Talpur, N. Systematic Literature Review of Information Extraction From Textual Data: Recent Methods, Applications, Trends, and Challenges. *IEEE Access* **2023**, *11*, 10535–10562. [[CrossRef](#)]
35. Steiner, T.; Verborgh, R.; Troncy, R.; Gabarro Valles, J.; Walle, R. Adding realtime coverage to the Google Knowledge Graph. In Proceedings of the 11th International Semantic Web Conference, Boston, MA, USA, 11–15 November 2012; pp. 1–4.
36. Regalia, B.; Janowicz, K.; Mai, G.; Varanka, D. GNIS-LD: Serving and Visualizing the Geographic Names Information System Gazetteer as Linked Data. In Proceedings of the Semantic Web, Heraklion, Greece, 3–7 June 2018; Springer: Cham, Switzerland, 2018.
37. Wang, P.; Li, S.; Sun, G.; Wang, X.; Chen, Y.; Li, H. RC-NVM: Enabling Symmetric Row and Column Memory Accesses for In-Memory Databases. In Proceedings of the IEEE International Symposium on High Performance Computer Architecture (HPCA), Vienna, Austria, 24–28 February 2018; pp. 518–530.
38. Bulla, M.; Hillebrand, L.; Lübbing, M.; Sifa, R. Knowledge Graph Based Question Answering System for Financial Securities. In Proceedings of the KI 2021: Advances in Artificial Intelligence, Virtual Event, 27 September–1 October 2021; Edelkamp, S., Möller, R., Rueckert, E., Eds.; Springer International Publishing: Cham, Switzerland, 2021; pp. 44–50.
39. Li, X.; Shu, H.; Zhai, Y.; Lin, Z. A Method for Resume Information Extraction Using BERT-Bi-LSTM-CRF. In Proceedings of the 2021 IEEE 21st International Conference on Communication Technology (ICCT), Tianjin, China, 13–16 October 2021; pp. 1437–1442.
40. Jin, X.; Zhang, S.; Liu, J. Word Semantic Similarity Calculation Based on Word2vec. In Proceedings of the 2018 International Conference on Control, Automation and Information Sciences (ICCAIS), Hangzhou, China, 24–27 October 2018; pp. 12–16.

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.