



An error recognition method for power equipment defect records based on knowledge graph technology

Hui-fang WANG^{†‡}, Zi-quan LIU

College of Electrical Engineering, Zhejiang University, Hangzhou 310027, China

[†]E-mail: huifangwang@zju.edu.cn

Received Apr. 24, 2018; Revision accepted Sept. 14, 2018; Crosschecked Nov. 12, 2019

Abstract: To recognize errors in the power equipment defect records in real time, we propose an error recognition method based on knowledge graph technology. According to the characteristics of power equipment defect records, a method for constructing a knowledge graph of power equipment defects is presented. Then, a graph search algorithm is employed to recognize different kinds of errors in defect records, based on the knowledge graph of power equipment defects. Finally, an error recognition example in terms of transformer defect records is given, by comparing the precision, recall, F_1 -score, accuracy, and efficiency of the proposed method with those of machine learning methods, and the factors influencing the error recognition effects of various methods are analyzed. Results show that the proposed method performs better in error recognition of defect records than machine learning methods, and can satisfy real-time requirements.

Key words: Error recognition; Power equipment defect record; Knowledge graph; Machine learning
<https://doi.org/10.1631/FITEE.1800260>

CLC number: TM76; TP181

1 Introduction

There has been much literature concerning equipment defects recorded during routine power equipment inspection (Radeva et al., 2009; Zheng and Dagnino, 2014). A defect record, such as “The oil temperature of the transformer’s tank is too high and reaches 98 degrees” describes the defect component, phenomenon, and quantitative information about the degree of defect. As first-hand materials, these records are not only the foundation of defect classification and elimination, but also directly related to the accuracy of health condition evaluation and power equipment maintenance decisions (Qiu et al., 2015). However, due to the limited knowledge and experience of inspectors, human errors occur frequently in defect records, such as omissions and contradictions

(Dhillon and Liu, 2006), which can affect defect treatment, equipment condition evaluation, and other subsequent work.

With the popularization of handheld mobile intelligent terminals for recording defects, if an error recognition function is added to the terminals, the terminals can give a prompt when incorrect records are entered, and the quality of defect records can be guaranteed from the source.

For different error conditions in defect/fault text in the power industry, several recognition methods have been proposed. Rudin et al. (2012) used redundant information in a trouble ticket to detect contradictory descriptions. Furthermore, Rudin et al. (2014) dealt with the problem of conflicting descriptions by combining overlapping information, domain expertise, and descriptive statistics. Liddy et al. (2013) applied manual annotation to identify common misspellings and informal names. Similarly, Qiu et al. (2016) recommended specific unlisted words to recognize nonstandard descriptions in defect records. Aiming at the lack of key information in some defect

[‡] Corresponding author

ORCID: Hui-fang WANG, <http://orcid.org/0000-0002-1483-364X>

© Zhejiang University and Springer-Verlag GmbH Germany, part of Springer Nature 2019

records, Cao et al. (2017) defined a power semantic framework and discovered information omissions in the process of filling slots.

However, when these methods are applied to error recognition in power equipment defect records, they have two shortcomings as follows:

1. Power equipment defect records are short texts with limited information, most of which do not contain redundant information. Thus, it is almost impossible to use information redundancy in a single text.

2. The complexity of power equipment structure, defects, and colloquial phenomena in defect records result in diverse texts, making it difficult to comprehensively consider the error conditions of defect records by manual analysis (Devaney et al., 2005; Rudin et al., 2012) or even power industry specifications based on expert knowledge (Huang and Zhou, 2015).

To overcome these shortcomings, a feasible idea is to learn the rules from existing defect records using machine learning methods, and apply the rules to error recognition in new records, to avoid dependence on redundant information and human knowledge. In fact, some researchers have studied the application of machine learning to power equipment defect records (Xie et al., 2016; Wei et al., 2017; Liu et al., 2018). However, machine learning is data-driven and difficult to interpret. If used for error recognition in defect records, machine learning methods not only are influenced by the characteristics of training data, but also probably ignore key information because the methods cannot use reasoning to evaluate defect records that appear to be both correct and incorrect because of their similarity.

Another feasible idea is based on the knowledge graph technology, which constructs a knowledge graph of power equipment defects from the existing defect records and identifies errors in new records with the aid of the knowledge graph. Formally proposed by Amit (2012), a knowledge graph is a knowledge network connecting entities and properties through relations. Its basic unit is triples of “entity-relation-entity” or “entity-relation-property,” while the entities and properties exist as nodes and the relations are presented as directed edges connecting two nodes. With the interpretable graph structure, a knowledge graph can express the complex relationships among information contained in texts, making it

possible to identify key information by knowledge reasoning.

Domains of knowledge graphs can be open or enclosed. Open-domain knowledge graphs are not confined to the field of knowledge. They require a wide range of knowledge coverage, and are used mainly for search engines with limited application depth (Bollacker et al., 2007; Suchanek et al., 2008; Bizer et al., 2009). On the contrary, enclosed-domain knowledge graphs can be applied only to specific industries. Because entities, properties, and relations are highly specialized and can be listed according to demands, the application of enclosed-domain knowledge graphs can be deeper and more targeted. Although enclosed-domain knowledge graphs have not been applied to power equipment defect records, they have been widely used in medicine (Goodwin and Harabagiu, 2013; Rotmensch et al., 2017; Shi et al., 2017), economics (Hu et al., 2017; Pujara, 2017), and other industries.

2 Automatic construction of a knowledge graph of power equipment defects

2.1 General process of knowledge graph construction

The general process of knowledge graph construction is divided mainly into three steps: knowledge extraction, knowledge fusion, and knowledge processing (Liu et al., 2016).

The purpose of knowledge extraction is to extract entities, properties, and relations from unstructured data as the basic elements of the knowledge graph.

In the process of knowledge fusion, entity disambiguation and coreference resolution are first applied to entities. Entity disambiguation discriminates entity names with multiple meanings (e.g., “apple” may refer to a fruit or a company name), while coreference resolution merges nouns and pronouns referring to the same thing into a node. Then, entities, properties, and relations are integrated to form a knowledge graph, along with the existing structured data.

Knowledge processing is a dynamic operation that evaluates the quality of the knowledge graph in subsequent application, and that updates the

knowledge graph according to the development of knowledge.

2.2 Construction process of a knowledge graph of power equipment defects

A power equipment defect record usually exists in the form of a single sentence, which generally describes the defect component, phenomenon, degree, and other related information in natural language. Considering the characteristics of power equipment defect records, we propose the following modifications based on the general process of knowledge graph construction:

1. In power equipment defects, as properties of defect components, defect phenomena may have properties like defect degrees. Thus, in addition to extracting relations between entities and between entities and properties, it is necessary to extract the relations between properties.

2. The meanings of entities are limited to the field of power equipment, which has clear specifications, so the step of entity disambiguation can be obviated.

3. Synonyms appear in the properties. Therefore, coreference resolution should be applied to properties apart from entities. Moreover, because the amount of data in the enclosed domain is relatively small, coreference resolution should be carried out before relation extraction, which will contribute to many training samples for extracting relations.

4. Specification of power equipment defects summarizes part of triples in tabular form, which can be used in the training process of relation extraction to make full use of structured data.

5. After being extracted, relations need to be screened to avoid relation redundancies that affect the application of knowledge graph.

6. In the data integration step, entities, properties, and relations extracted from the unstructured data are combined with the triples contained in the specification, forming a knowledge graph of power equipment defects.

Fig. 1 shows the modified process of knowledge graph construction. The data integration method is consistent with that of the general construction process, while other steps need to be specially designed, which will be explained below.

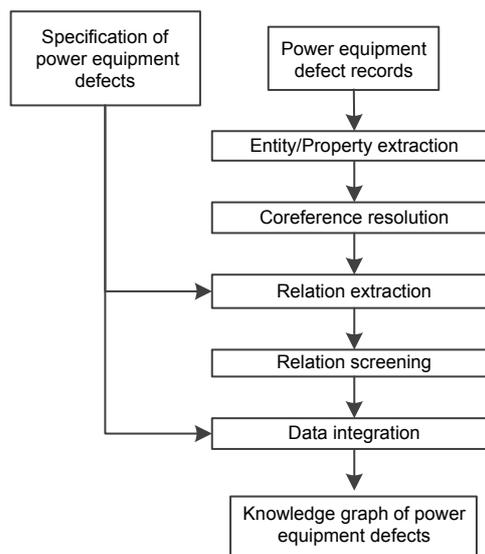


Fig. 1 Modified process of knowledge graph construction of power equipment defects

2.3 Entity/Property extraction

The main task of entity/property extraction is to extract the words representing entities/properties in power equipment defect records and carry out part-of-speech (POS) tagging. Because entities and properties can be listed exhaustively, the power industry dictionary is used to extract them (IEC, 2014; Liu et al., 2018). The specific steps are as follows:

1. Word segmentation. This is a necessary step for Chinese texts, in which there is no space between words. A word segmentation tool, “jieba,” is adopted to split the records into words (Lv, 2015). In this way, the dictionary of common words and the power industry dictionary are used to preliminarily match and segment the words, and then the hidden Markov model is used to recognize words that are outside the dictionaries (Baum and Petrie, 1966). For English or other languages where there is a space between words, this step can be skipped.

2. Word extraction. Search each word of the defect records in the power industry dictionary. If a word can be found in the dictionary, extract the entity/property represented by the word as an element of the knowledge graph.

3. POS tagging. Tag POSs of all the words according to the POSs of the words in the dictionary of common words and the power industry dictionary, and divide all the words into five categories: (1)

nouns describing power equipment and components, which are tagged as “En” (prefix “E” stands for entity); (2) verbs describing defect phenomena, which are tagged as “Pv” (prefix “P” stands for property); (3) adverbs describing the defect degree, which are tagged as “Pad;” (4) quantifiers describing the defect degree, which are tagged as “Pq;” (5) words that cannot be found in the power industry dictionary and that do not represent entities or properties are tagged according to POSs in the dictionary of common words.

When considering languages other than Chinese, typically English, some details of the method mentioned above need to be modified. For example, besides words, some phrases that represent entities/properties should be extracted. Meanwhile, a word may have several inflectional forms, which should be considered and unified.

2.4 Coreference resolution

Because power equipment defect records contain few pronouns, the coreference resolution needs only to consider synonyms of words that represent entities/properties. The steps are as follows:

1. Categorize words by their POSs. The POSs of synonyms must be the same. So, words that represent entities/properties can be divided into four groups according to their POSs, and synonym recognition is separately applied to each group.

2. Vectorize the words. To describe the semantic similarity between words that represent entities/properties, a word2vec method is used to train the words in defect records and transfer them into vectors (Mikolov et al., 2013). Then, by calculating the cosine similarity between word vectors, the semantic similarity between words can be judged.

3. Screen word pairs. When the words are vectorized, the words adjacent to each other in a sentence (adjacent word pair) tend to have high cosine similarity, and the same goes for the words with similar contexts in different sentences (appositive word pairs) (Grover and Leskovec, 2016). However, only the appositive word pairs can be synonyms, which are almost impossible to appear in the same defect record. Thus, we screen out the word pairs whose words have ever appeared in the same record to remove the adjacent word pairs.

4. Form a list of synonyms. Merge the appositive word pairs that include the same word into a set of

synonyms. Select a word from each set as the standardized name representing all the words in the set, and express the synonym set in the form of a list.

2.5 Relation extraction

The main task of relation extraction is to identify whether there are relations and what relations exist between any two entities/properties. In the knowledge graph of power equipment defects, the types of relations can be defined according to the POSs of two entities/properties (Table 1).

Table 1 Relation types of two entities/properties

POS		Possible relations between p and q
p	q	
En	En	p contains q ; q contains p ; no relation
En	Pv	q is a defect phenomenon of p ; no relation
Pv	Pad	q is a qualitative description of p ; no relation
Pv	Pq	q is a quantitative description of p ; no relation

As a result, the relation extraction is transformed into a classification problem, and the training set is provided by specification of power equipment defects. Because the number of training samples is relatively small, which may affect the performance of supervised training, it is necessary to apply semi-supervised cooperative training to relation classification.

Before classification, among all the word pairs formed by words representing entities/properties, select the word pairs belonging to the four POS combinations in Table 1, which will be classified according to Algorithm 1. The statements with an asterisk will be explained in detail.

Algorithm 1 Relation classification

Input: word pairs, defect records, and n^*

Output: relations (containing relations of word pairs)

1 train_pairs={the word pairs whose relations are defined in the specification}

2 predict_pairs={the word pairs whose relations are not defined in the specification}

3 pairs={all the word pairs}

4 records={all the defect records}

5 Initialize train_instances, predict_instances, and train_labels to empty sets

6 **for** $i=1$ to the number of elements in train_pairs **do**

```

7  for  $j=1$  to the number of elements in records do
8    if records[ $j$ ] contains two words in train_pairs[ $i$ ] then
9      Append records[ $j$ ] to train_instances
10     Append the relation of train_pairs[ $i$ ] to train_labels
11   end if
12 end for
13 end for
14 for  $i=1$  to the number of elements in predict_pairs do
15   for  $j=1$  to the number of elements in records do
16     if records[ $j$ ] contains two words in predict_pairs[ $i$ ]
17       then
18       Append records[ $j$ ] to predict_instances
19     end if
20   end for
21 while predict_instances is not empty do
22   vectors_t1={vectorized train_instances by method1*}
23   vectors_t2={vectorized train_instances by method2*}
24   vectors_p1={vectorized predict_instances by method1}
25   vectors_p2={vectorized predict_instances by method2}
26   Train classifier1 with vectors_t1 and train_labels
27   Train classifier2 with vectors_t2 and train_labels
28   Predict vectors_p1 with classifier1
29   Predict vectors_p2 with classifier2
30   if the number of elements in predict_instances >  $n$  then
31     vecs_p1={ $n$  vectors with the highest predicted prob-
32     ability in vectors_p1}
33     vecs_p2={ $n$  vectors with the highest predicted prob-
34     ability in vectors_p2}
35   else
36     vecs_p1={all the vectors in vectors_p1}
37     vecs_p2={all the vectors in vectors_p2}
38   end if
39   instances_p1={corresponding instances of vecs_p1}
40   instances_p2={corresponding instances of vecs_p2}
41   instances_p={merged instances_p1 and instances_p2}
42   labels_p={corresponding labels of instances_p*}
43   Append all the instances in instances_p to train_
44   instances
45   Append all the labels in labels_p to train_labels
46   Delete instances in instances_p from predict_instances
47 end while
48 for  $i=1$  to the number of elements in pairs do
49   Initialize labels to empty sets
50   for  $j=1$  to the number of elements in records do
51     if train_instances[ $j$ ] contains two words in pairs[ $i$ ] then
52       Append train_labels[ $j$ ] to labels
53     end if
54   end for
55   Append the most label in labels to relations
56 end for

```

Details of Algorithm 1 are as follows:

1. n is one of the parameters of semi-supervised cooperative training, which has little influence on the training effect in a proper range and can be

determined by experiments (Chen et al., 2013).

2. The relation between two words in a word pair is not only related to the relative position and meanings of the two words, but also relevant to the number, POSs, and meanings of the words between the two words (Li et al., 2008). Thus, two methods to vectorize the instances of a word pair are as follows: method1 selects the relative position of the two words and the number and POSs of the words between the two words as features, as shown in Table 2; method2 takes the word vectors in Section 2.4 as features to reflect the meanings of the words. First, set the maximum number of words between the two words of all the instances as v , and there are totally $(v+2)$ words along with the two words. Then, splice the $(v+2)$ word vectors to form the vector of the corresponding instance. For the other instances, in which the number of the words between the two words is smaller than v , fill in the vacant features with zero.

Table 2 Features of method1

No.	Description
1	Value=1 when A is in front of B ; value=0 when B is in front of A
2	The number of words with POS "En" between A and B
3	The number of words with POS "Pv" between A and B
4	The number of words with POS "Pad" between A and B
5	The number of words with POS "Pq" between A and B
6	The number of words with POS "conj" between A and B
7	The number of words with POS "loc" between A and B
8	The number of punctuation marks between A and B
9	Total number of words between A and B

A and B represent two words in a word pair

3. After merging instances_p1 and instances_p2, if an instance in instances_p is contained in both instances_p1 and instances_p2, its corresponding label will be the one with a higher predicted probability.

The method mentioned above can extract the relation contained in the Chinese texts. For other languages (like English), with a strict grammar, a semantic framework can be defined by analyzing the construction of the records (Cao et al., 2017). By filling the words of the records in the slots of the framework, relations between words that represent

entities/properties can be extracted according to the slot positions.

2.6 Relation screening

Relation screening eliminates redundant containing relations. Inspectors usually do not strictly record the defect components rank-by-rank according to the specifications; for example, “变压器冷却器系统风扇故障” (“the fan in the cooling system of the transformer broke down”) may be recorded as “变压器风扇故障” (“the fan of the transformer broke down”). As a result, although “变压器” (“transformer”) does not directly contain “风扇” (“fan”), they are probably recognized as having a containing relation, which leads to the structure in Fig. 2.

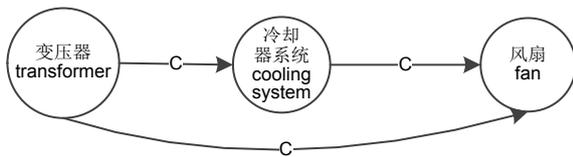


Fig. 2 An example of containing relations

An edge with “C” means the relation between p and q (p points to q) is “ p contains q ”

In Fig. 2, the containing relation between “变压器” (“transformer”) and “风扇” (“fan”) can be recognized by knowledge reasoning. If all the indirect containing relations are expressed, the complexity of the knowledge graph will be greatly increased. Thus, if there is another path connecting two entities that have a containing relation, the containing relation between the two entities will be eliminated. For example, there is a path “变压器—冷却器系统—风扇” (“transformer – cooling system – fan”) between “变压器” (“transformer”) and “风扇” (“fan”), so the edge representing the containing relation between “变压器” (“transformer”) and “风扇” (“fan”) will be eliminated.

2.7 Knowledge graph updating

To update the knowledge graph, there are two situations to be considered.

If there is abundant knowledge updating at the same time, such as substantial updating of the dictionaries or a rapid increase of records containing new types of defects, a direct way is to repeat the

construction process in Section 2.2 with updated dictionaries or records in the corresponding steps.

However, in practice, knowledge update is usually gradual. Thus, the limited number of new words or records cannot provide sufficient training samples, and repetition of the construction process is probably ineffective. Thanks to the good interpretability, manual modification can be applied to the constructed knowledge graph, which can ensure the accuracy and will not take much time because the updating is gradual. In this case, apart from the new entities/properties/relations that explicitly appear in the dictionaries, new records can be resolved into several entities/properties/relations. Therefore, in essence, the updating of the knowledge graph is to add and modify corresponding nodes (representing entities/properties) and edges (representing relations) according to the updated knowledge based on expertise.

3 Error recognition of power equipment defect records

3.1 Error types in power equipment defect records

A standard defect record should completely describe one defect; that is, it must include a clear defect subject and its corresponding defect phenomenon, and may also include qualitative and quantitative descriptions of defect degree (Qiu et al., 2016).

In the process of recording power equipment defects, the following types of errors may occur due to the limited knowledge and experience of inspectors:

1. The defect subject is not clear.

(1) The defect subject is missing; that is, the defect record does not contain any entity. So, it is impossible to find the defect subject.

(2) The entity information is ambiguous. As mentioned in Section 2.6, inspectors usually do not record the defect components rank-by-rank, so the missing entities need to be inferred from the recorded entities. However, if the key entity information is missing, it will be impossible to infer the defect subject because of ambiguity. For example, in the record “变压器呼吸器硅胶变色” (“the silica gel in the breather of the transformer changed color”), the breather may be a part of the transformer’s main body

or a part of the transformer's on-load tap-changer (OLTC), which cannot be determined from the recorded information.

(3) The entity information is contradictory. If some of the entities are incorrectly recorded, it will lead to contradictions in entity information and a confusing defect subject. For example, in the record “变压器无载开关呼吸器硅胶变色” (“the silica gel in the breather of the transformer's off-circuit tap-changer changed color”), because there is no breather installed on the off-circuit tap-changer, the information concerning the breather and the off-circuit tap-changer is contradictory.

2. The defect phenomenon is incorrectly recorded.

(1) The defect phenomenon is missing. That is, no information about defect phenomena is recorded.

(2) Several defect phenomena are recorded. Different defect phenomena should be recorded separately, so that different methods can be used to deal with them. Thus, each defect record should contain only one defect phenomenon.

(3) The defect phenomenon does not correspond to the defect subject. That is, the recorded phenomenon cannot possibly occur for the recorded subject.

3. The qualitative and quantitative descriptions of the defect degree are incorrectly recorded. Qualitative and quantitative descriptions are not necessary in a defect record. However, if they are recorded, they should correspond to the defect phenomenon; otherwise, the meaning of the record will be confusing.

3.2 Graph search algorithm for error recognition

When checking if there are any of the above three types of errors in a defect record, word segment and POS tagging are first applied to the record. Next, according to the list of synonyms, replace the words in the defect record with a standardized name, and mark the nodes of the knowledge graph that correspond to the entities/properties appeared in the defect record. Error recognition of the defect record is then performed in Fig. 3.

The knowledge graph in Fig. 4 illustrates the error recognition process based on a graph search. Suppose that the marked nodes are the gray nodes as shown in Fig. 4.

According to Fig. 4, the subsequent steps are as follows:

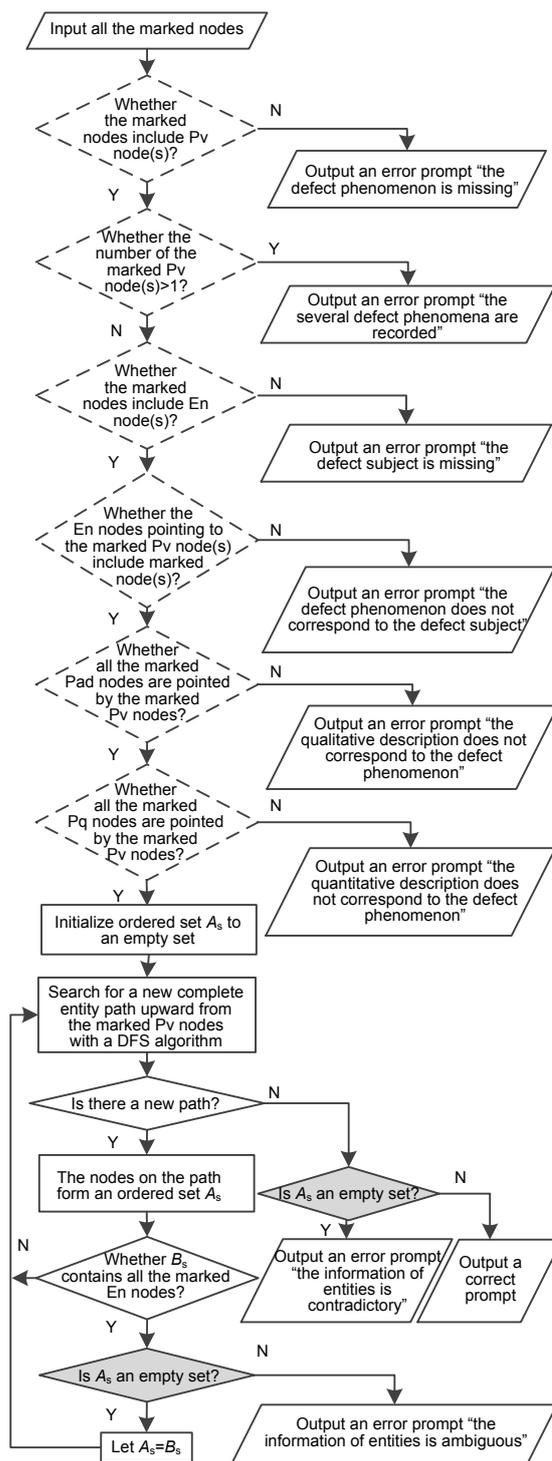


Fig. 3 Process of error recognition of a defect record

1. After all the marked nodes are input, check the first six decision boxes (boxes with dotted frame lines and white bottoms in Fig. 3). Because node j is marked, the first decision box is determined to be

“yes” and does not create any error prompt. Likewise, none of the other five decision boxes create any error prompt.

2. Let ordered set A_s be an empty set.

3. Search for a path upward from node j with the depth-first search (DFS) algorithm, so it will search for the edges $10 \rightarrow 1 \rightarrow 2 \rightarrow 3$ and pass the nodes $i \rightarrow f \rightarrow e \rightarrow g$, thus forming an ordered set $\{i, f, e, g\}$ as set B_s , which does not contain all the marked entity nodes.

4. Continue searching for another path. According to the DFS algorithm, the path $10 \rightarrow 1 \rightarrow 2 \rightarrow 4 \rightarrow 5$ will be searched, so B_s is $\{i, f, e, b, a\}$, which does not contain all the marked entity nodes.

5. Continue searching for another path. The path $10 \rightarrow 1 \rightarrow 6 \rightarrow 7 \rightarrow 8 \rightarrow 5$ will be searched, so B_s is $\{i, f, d, c, b, a\}$, which contains all the marked entity nodes. Meanwhile, A_s is an empty set. So, let $A_s = B_s$.

6. Continue searching until there is no unsearched path. Because A_s is not an empty set here, a correct prompt will be output.

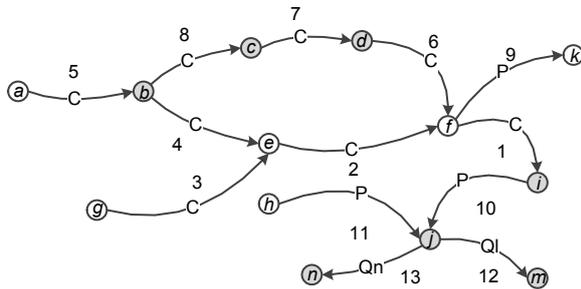


Fig. 4 An example of the knowledge graph

Nodes $a-i$ represent entities with POS “En,” nodes j and k represent properties with POS “Pv,” node m represents property with POS “Pad,” and node n represents property with POS “Pq.” An edge with “C” means that the relation between p and q (p points to q , the same as below) is “ p contains q ,” an edge with “P” means that the relation between p and q is “ q is a defect phenomenon of p ,” an edge with “Ql” means that the relation between p and q is “ q is a qualitative description of p ,” and an edge with “Qn” means that the relation between p and q is “ q is a quantitative description of p ”

According to the above process of graph search, although nodes a and f are not marked, it can be inferred from ultimate A_s that the defect record implies the information of nodes a and f , which is the process of knowledge reasoning.

Additionally, if nodes c and d are not marked, it

will be impossible to judge whether the entity represented by node f comes from the entity of node d or that of node e . In this situation, according to Fig. 3, after A_s becomes a non-empty set, it will still come to the decision box with a dotted frame line and gray bottom, so the error prompt “the information of entities is ambiguous” will be given. If node b is not marked and node g is marked, nodes c and d will not correspond to node g . According to Fig. 3, A_s will be a null set after searching for all the paths, and then it will come to the decision box with a solid frame line and gray bottom and give the error prompt “the information of entities is contradictory.”

4 Case study

4.1 Data source, models, and indices

To study the error recognition effect of the proposed method, an experiment was performed based on transformer defect records. A total of 7596 transformer defect records recorded in 2013–2015 from a Chinese power grid company were selected, including 6848 correct and 748 incorrect manually tagged records. Then, 3424 correct records and 374 incorrect records were randomly selected as a training set, while the remaining 3424 correct records and 374 incorrect records constituted a test set. When constructing the transformer defect knowledge graph, the correct training set records were used as the source of unstructured data, and the relation extraction training set was provided by specification Q/GDW 1904.1-2013 (Q/GDW, 2013). No structured data information was added to the first knowledge graph model KG1, while information from structured data in the specification was added to the second knowledge graph model KG2.

In addition, compared with the knowledge graph models, four machine learning classifiers, namely, logistic regression (LR), modified linear support vector machine (MLSVM) (which can automatically adjust penalty factors according to the proportion of positive samples to negative samples), SVM with radial basis function (SVMR), and weighted random forest (WRF) (which can automatically adjust class weights according to the proportion of positive samples to negative samples), were used as controls and achieved with the scikit-learn toolkit in Python. After

significantly better in recall and F_1 -score. Although the precision of LR is the highest, it does not mean that LR has good performance. If a model correctly recognizes only one incorrect record in the test set and predicts all the other records to be correct, its precision is 100%; however, the prediction result indicates an obviously poor ability to identify incorrect records. Therefore, the recall and F_1 -score of the model will be very low. In addition, the model accuracy is generally high, mainly because of the high proportion of correct records. Even if a model predicts that all the records are correct, the accuracy can reach about 90% (the proportion of the correct records). In this case, the advantages of KG1 and KG2 in accuracy benefit from their superior ability to recognize incorrect records.

Table 3 Statistical results of different models

Model	P (%)	R (%)	F (%)	A (%)	t (ms)
LR	100.00	20.86	34.51	92.21	212.69
MLSVM	70.06	64.44	67.13	93.79	3879.63
SVMR	73.97	62.30	67.63	94.13	6975.86
WRF	70.10	72.73	71.39	94.26	349.13
KG1	87.50	97.33	92.15	98.37	7885.31
KG2	88.19	97.86	92.78	98.50	10 228.56

In practical applications, the error recognition function is used to give a prompt when incorrect records are entered. Therefore, high recalls of the knowledge graph models mean that they can identify most of the incorrect records and give prompts. Although the precision is nearly 90%, which means that about 10% of correct records will be recognized as incorrect records, they can be manually changed to correct records after error prompts are given. On the contrary, low recalls of the machine learning methods mean that many incorrect records will be judged as correct records, while error prompts will not be given. Thus, these records cannot be corrected.

Because the error recognition algorithm of the knowledge graph models is directed against the defect record error types, the knowledge graph models can output error type when giving an error prompt, whereas the machine learning methods can give only a prompt without an error type.

From the comparison of KG1 and KG2 in Table 3, it can be seen that the added structured data can supplement triple information and optimize the knowledge graph structure, which will improve the

error recognition effect of the knowledge graph model. Although the total test time of KG2 is the longest, the average time spent on each record is only about 2.7 ms (the number of records in the test set is 3798); that is, error recognition results can be given in about 2.7 ms after a defect record is entered in the practical application, which can satisfy the real-time requirement.

There are two main reasons why the knowledge graph models are superior to machine learning models in the comprehensive performance of error recognition:

1. The influence of training data characteristics

One of the characteristics of the training set is that the data is significantly skewed (the ratio of correct records to incorrect records is about 9:1), which causes the machine learning methods to tend to predict the records in the test set as correct records (Lampert and Gañarski, 2014). When the training set is given, the data skewness can be reduced only by removing some correct records, which will lead to reduction of another characteristic of the training set, i.e., the total number of defect records. On the other hand, the construction of the knowledge graph depends only on the correct records and is not affected by the content and quantity of the incorrect records.

To analyze the influence of the training data characteristics, the original training set was used to carry out further experiments. By successively removing some correct records in the training set, ratios of correct records to incorrect records become 7:1, 5:1, 3:1, and 1:1. Then, each model was trained and applied to recognize the incorrect records in the original test set. Statistical results are shown in Table 4.

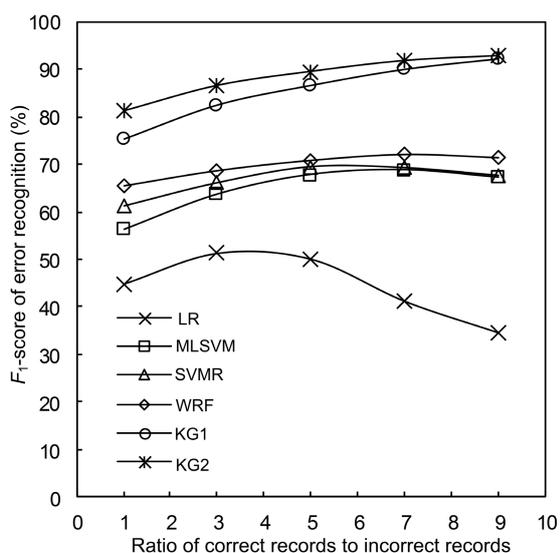
According to Tables 3 and 4, F_1 -scores of the six models under different data skewness are shown in Fig. 6.

In Fig. 6, the F_1 -score of LR is sensitive to data skewness, and becomes small with the increase of the skewness when the ratio of correct records to incorrect records is larger than three. While the F_1 -scores of MLSVM and WRF are influenced mainly by the total amount of training data, and become small when the ratio of correct records to incorrect records is larger than seven. Because the influence of training data characteristics is uncertain, machine learning models cannot achieve the optimal effect if the data skewness and the number of the training records are

Table 4 Statistical results of different models under different data skewness

Model	Ratio	<i>P</i> (%)	<i>R</i> (%)	<i>F</i> (%)	<i>A</i> (%)
LR	1:1	34.12	65.51	44.87	84.15
	3:1	94.29	35.29	51.36	93.42
	5:1	100.00	33.42	50.10	93.44
	7:1	100.00	25.94	41.19	92.71
MLSVM	1:1	47.50	68.72	56.17	89.44
	3:1	55.73	74.06	63.61	91.65
	5:1	63.19	72.99	67.74	93.15
	7:1	68.72	68.72	68.72	93.84
SVMR	1:1	51.76	75.07	61.27	90.68
	3:1	58.25	76.47	66.13	92.29
	5:1	64.60	75.13	69.47	93.50
	7:1	65.02	74.06	69.25	93.52
WRF	1:1	54.91	80.75	65.37	91.57
	3:1	59.22	81.55	68.62	92.65
	5:1	64.69	79.14	70.81	93.58
	7:1	67.52	77.27	72.07	94.10
KG1	1:1	68.72	83.42	75.36	94.63
	3:1	76.07	90.11	82.50	96.23
	5:1	80.79	93.32	86.60	97.16
	7:1	84.83	95.72	89.95	97.89
KG2	1:1	76.11	86.90	81.15	96.02
	3:1	81.86	91.71	86.51	97.18
	5:1	85.23	94.12	89.45	97.81
	7:1	87.41	96.52	91.74	98.29

Ratio: the ratio of correct records to incorrect records

**Fig. 6 F_1 -score curves of different models under different data skewness**

not properly selected. On the contrary, the construction of the knowledge graph depends only on correct records. More correct records will provide more effective information, which helps improve the integrity and accuracy of the knowledge graph. Thus, the influence of the training data characteristics on the knowledge graph models is deterministic; that is, the error recognition effect becomes better as the number of correct records in the training set increases.

In practice, the skewness of defect records is unavoidable, because with relatively standardized management of defect records, there tend to be far fewer incorrect records than correct records. Moreover, with further regulation of defect record management, the proportion of incorrect records will decrease. So, the data skewness will increase, and the advantage of the knowledge graph models will be more obvious.

2. The ability to recognize key information

Because the knowledge graph models can realize knowledge reasoning aiming at error types, they can recognize key information that determines the correctness of a defect record, whereas the machine learning models more easily ignore key information in a relatively long defect record. Two defect records were used to give an intuitive explanation (Table 5). Both A1 and A2 are records in the original test set, and the judgments of LR, MLSVM, and KG2 are all in agreement with reality.

If we delete the word “有载开关” (“OLTC”) in A1, then it cannot be determined if the defective component is the gas relay of the transformer’s main body or the gas relay of the transformer’s OLTC, and A1 will become an incorrect record (denoted as B1). A2 is an incorrect record because there is no oil-level indicator on the tank. Replacing the word “油箱” (“tank”) with “储油柜” (“conservator”), A2 will become a correct record (denoted as B2). Then LR, MLSVM, and KG2 were used to judge the correctness of B1 and B2, and the results are shown in Table 6.

In Tables 5 and 6, the judgments of LR and MLSVM have not changed in response to the change of the defect records, while KG2 has adjusted its judgment according to the change of the key information. For machine learning models, as for the features of a sentence, all the words in a defect record will influence the judgment together. For example,

because most of the records about oil leaks of gas relays in the training set are correct records, when the words “变压器” (“transformer”), “气体继电器” (“gas relay”), “漏油” (“leaking oil”), and “滴” (“drop”) appear at the same time, the machine learning models are likely to judge both A1 and B1 as correct records according to the word features. Even if the word “有载开关” (“OLTC”) is deleted, its influence may be overwhelmed by other words, which will result in misjudgment.

By contrast, the change of key information can be visually reflected in the knowledge graph, so the correct judgment can be obtained by knowledge reasoning (Fig. 7). When A1 is changed to B1, the reflection of the defect record in the knowledge graph changes from Fig. 7a to Fig. 7b. There are two alternative paths from “变压器” (“transformer”) to “气体继电器” (“gas relay”), and it is impossible to determine which is the real path by the marked nodes. Therefore, according to the process in Fig. 3, the error prompt “the information of entities is ambiguous” will be output. When A2 is changed to B2, the reflection of the defect record in the knowledge graph changes from Fig. 7c to Fig. 7d. In Fig. 7c, there is no path containing all the marked entity nodes, so the error prompt “the information of entities is contradictory” will be output. While there is such a path in Fig. 7d, which is highlighted by the bold edges, the defect record will be judged to be correct.

5 Conclusions

We have introduced knowledge graph technology into error recognition of power equipment defect records, and proposed an error recognition method based on knowledge graph technology. In this study, the construction of a knowledge graph of power equipment defects has been described in detail. Based on this, the graph search algorithm has been proposed to recognize errors in power equipment defect records. Results and analysis of the examples indicated the remarkable superiority in recognition effect and the feasibility in efficiency of the knowledge graph model. So, the error prompt can be properly given in real time when an incorrect defect record is entered by the inspector, which will help ensure the quality of defect records from the source.

Apart from error recognition of defect records, we have provided a new idea for other text processing tasks in power systems, such as automatic severity grading of defect records and automatic retrieval of similar records for defect treatment suggestion. According to the research approach of this study, tasks can be processed by constructing a power knowledge graph first, and then designing an algorithm, based on the constructed knowledge graph, toward a specific demand. Therefore, the model design in the study is an important reference and demonstration for text mining in power systems.

Table 5 Two examples of defect records and the judgments

Defect record	Content	Correctness in reality	Judgment		
			LR	MLSVM	KG2
A1	变压器/有载开关/气体继电器/出现/轻微/漏油/现象/, /15/秒/每/滴/。 (The gas relay of the transformer's OLTC has a phenomenon of leaking oil 15 seconds per drop.)	T	T	T	T
A2	变压器/本体/油箱/油位计/指针/断裂/, /需要/进行/更换/。 (The needle in the oil-level indicator on the tank of the transformer's main body snaps and needs to be replaced.)	F	F	F	F

Table 6 Modified defect records and the judgments

Defect record	Content	Correctness in reality	Judgment		
			LR	MLSVM	KG2
B1	变压器/气体继电器/出现/轻微/漏油/现象/, /15/秒/每/滴/。 (The gas relay of the transformer has a phenomenon of leaking oil 15 seconds per drop.)	F	T	T	F
B2	变压器/本体/储油柜/油位计/指针/断裂/, /需要/进行/更换/。 (The needle in the oil-level indicator on the conservator of the transformer's main body snaps and needs to be replaced.)	T	F	F	T

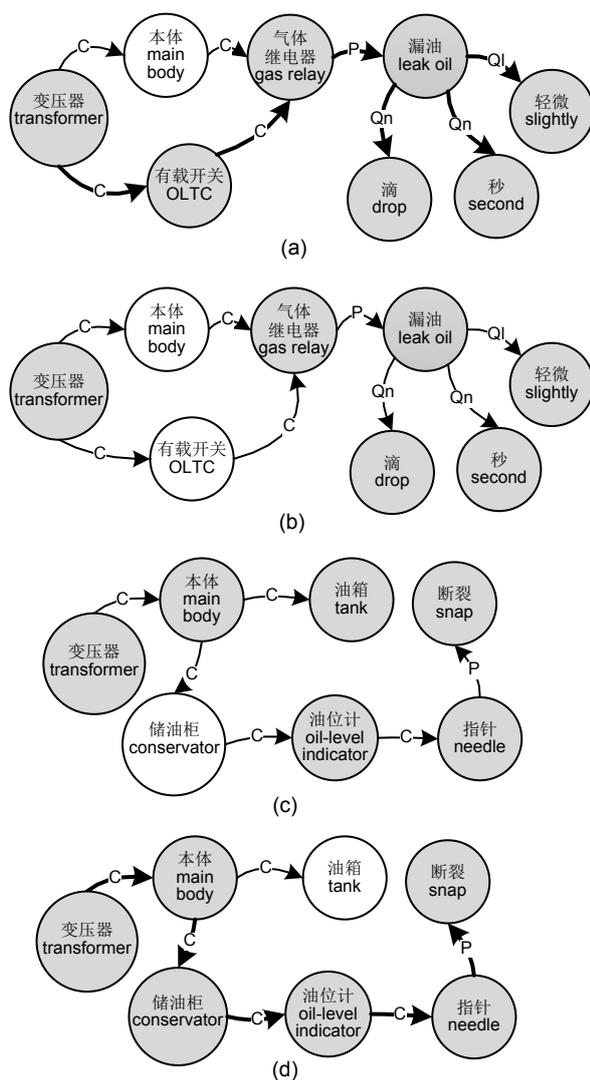


Fig. 7 Reflection of the defect records in the knowledge graph: (a) A1; (b) B1; (c) A2; (d) B2
 Gray nodes are the marked nodes corresponding to the defect records. The paths corresponding to the correct records are highlighted by bold edges

Because the accuracy of coreference resolution and relation extraction will directly influence the accuracy and integrality of knowledge graph, if more techniques of natural language processing like syntax parsing are used to extract more semantic features, the accuracies of coreference resolution and relation extraction may be further improved, which will benefit the error recognition effect of defect records. It is also an important direction in our further research.

Compliance with ethics guidelines

Hui-fang WANG and Zi-quan LIU declare that they have no conflict of interest.

References

- Amit S, 2012. Introducing the Knowledge Graph: Things, not Strings. <https://googleblog.blogspot.com/2012/05/introducing-knowledge-graph-things-not.html>
- Baum LE, Petrie T, 1966. Statistical inference for probabilistic functions of finite state Markov chains. *Ann Math Stat*, 37(6):1554-1563. <https://doi.org/10.1214/aoms/1177699147>
- Bizer C, Lehmann J, Kobilarov G, et al., 2009. DBpedia—a crystallization point for the Web of data. *J Web Semant*, 7(3):154-165. <https://doi.org/10.1016/j.websem.2009.07.002>
- Bollacker K, Cook R, Tufts P, 2007. Freebase: a shared database of structured general human knowledge. Proc 22nd National Conf on Artificial Intelligence, p.1962-1963.
- Cao J, Chen LS, Qiu J, et al., 2017. Semantic frame work-based defect text mining technique and application in power grid. *Power Grid Tech*, 41(2):637-643 (in Chinese). <https://doi.org/10.13335/j.1000-3673.pst.2016.1044>
- Chen LW, Feng YS, Zhao DY, 2013. Extracting relations from the Web via weakly supervised learning. *J Comput Res Dev*, 50(9):1825-1835 (in Chinese). <https://doi.org/10.7544/issn1000-1239.2013.20130491>
- Devaney M, Ram A, Qiu H, et al., 2005. Preventing failures by mining maintenance logs with case-based reasoning. Proc 59th Meeting of the Society for Machinery Failure Prevention Technology, p.1-10.
- Dhillion BS, Liu Y, 2006. Human error in maintenance: a review. *J Qual Mainten Eng*, 12(1):21-36. <https://doi.org/10.1108/13552510610654510>
- Goodwin T, Harabagiu SM, 2013. Automatic generation of a qualified medical knowledge graph and its usage for retrieving patient cohorts from electronic medical records. Proc IEEE 7th Int Conf on Semantic Computing, p.363-370. <https://doi.org/10.1109/icsc.2013.68>
- Grover A, Leskovec J, 2016. node2vec: scalable feature learning for networks. Proc 22nd ACM SIGKDD Int Conf on Knowledge Discovery and Data Mining, p.855-864. <https://doi.org/10.1145/2939672.2939754>
- Hu XB, Tang XH, Tang FL, 2017. Analysis of investment relationships between companies and organizations based on knowledge graph. Proc 11th Int Conf on Innovative Mobile and Internet Services in Ubiquitous Computing, p.208-218. https://doi.org/10.1007/978-3-319-61542-4_20
- Huang YH, Zhou XX, 2015. Knowledge model for electric power big data based on ontology and semantic web. *CSEE J Power Energy Syst*, 1(1):19-27. <https://doi.org/10.17775/cseejpes.2015.00003>
- IEC, 2014. International Electrotechnical Vocabulary (IEV): Generation, Transmission and Distribution of Electricity—Substations. International Electrotechnical Commission, Geneva.
- Lampert TA, Gançarski P, 2014. The bane of skew. *Mach Learn*, 97(1-2):5-32. <https://doi.org/10.1007/s10994-013-5432-x>

- Li WJ, Zhang P, Wei FR, et al., 2008. A novel feature-based approach to Chinese entity relation extraction. Proc 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies, p.89-92.
- Liddy DE, Symonenko S, Rowe S, 2013. Sublanguage analysis applied to trouble tickets. Proc 19th Int Florida Artificial Intelligence Research Society Conf, p.752-757.
- Liu Q, Li Y, Duan H, et al., 2016. Knowledge graph construction techniques. *J Comput Res Dev*, 53(3):582-600 (in Chinese).
<https://doi.org/10.7544/issn1000-1239.2016.20148228>
- Liu ZQ, Wang HF, Cao J, et al., 2018. A classification model of power equipment defect texts based on convolutional neural network. *Power Syst Technol*, 42(2):644-650 (in Chinese).
<https://doi.org/10.13335/j.1000-3673.pst.2017.1377>
- Lv SH, 2015. The Key Technology Research and Implementation of the Pinyin-to-Character Conversion System. MS Thesis, University of Electronic Science and Technology of China, Chengdu, China (in Chinese).
- Mikolov T, Chen K, Corrado G, et al., 2013. Efficient estimation of word representations in vector space.
<https://arxiv.org/abs/1301.3781>
- Pujara J, 2017. Extracting knowledge graphs from financial filings: extended abstract. Proc 3rd Int Workshop on Data Science for Macro—Modeling with Financial and Economic Datasets, p.1-2.
<https://doi.org/10.1145/3077240.3077246>
- Q/GDW, 2013. Defects Description Specification of Power Transmission and Substation Equipment, Part 1: Power Substation Equipments, Q/GDW 1904.1-2013. State Grid Corporation of China (in Chinese).
- Qiu J, Wang HF, Lin DY, et al., 2015. Nonparametric regression-based failure rate model for electric power equipment using lifecycle data. *IEEE Trans Smart Grid*, 6(2):955-964.
<https://doi.org/10.1109/TSG.2015.2388784>
- Qiu J, Wang HF, Ying GL, et al., 2016. Text mining technique and application of lifecycle condition assessment for circuit breaker. *Autom Electron Power Syst*, 40(6): 107-112 (in Chinese).
<https://doi.org/10.7500/AEPS20150812003>
- Radeva A, Rudin C, Passonneau R, et al., 2009. Report cards for manholes: eliciting expert feedback for a learning task. Proc Int Conf on Machine Learning and Applications, p.719-724. <https://doi.org/10.1109/icmla.2009.72>
- Rotmensch M, Halpern Y, Tlimat A, et al., 2017. Learning a health knowledge graph from electronic medical records. *Sci Rep*, 7(1):1-11.
<https://doi.org/10.1038/s41598-017-05778-z>
- Rudin C, Waltz D, Anderson RN, et al., 2012. Machine learning for the New York City power grid. *IEEE Trans Patt Anal Mach Intell*, 34(2):328-345.
<https://doi.org/10.1109/tpami.2011.108>
- Rudin C, Ertekin S, Passonneau R, et al., 2014. Analytics for power grid distribution reliability in New York City. *Interfaces*, 44(4):351-439.
<https://doi.org/10.1287/inte.2014.0748>
- Shi LX, Li SJ, Yang XR, et al., 2017. Semantic health knowledge graph: semantic integration of heterogeneous medical knowledge and services. *Biomed Res Int*, 2017:1-12. <https://doi.org/10.1155/2017/2858423>
- Suchanek FM, Kasneci G, Weikum G, 2008. YAGO: a large ontology from Wikipedia and WordNet. *J Web Semant*, 6(3):203-217.
<https://doi.org/10.1016/j.websem.2008.06.001>
- Wei DQ, Wang B, Lin G, et al., 2017. Research on unstructured text data mining and fault classification based on RNN-LSTM with malfunction inspection report. *Energies*, 10(3):1-22. <https://doi.org/10.3390/en10030406>
- Xie C, Zou GP, Wang HF, et al., 2016. A new condition assessment method for distribution transformers based on operation data and record text mining technique. Proc China Int Conf on Electricity Distribution, p.1-7.
<https://doi.org/10.1109/ciced.2016.7576179>
- Zheng J, Dagnino A, 2014. An initial study of predictive machine learning analytics on large volumes of historical data for power system applications. Proc IEEE Int Conf on Big Data, p.952-959.
<https://doi.org/10.1109/bigdata.2014.7004327>