Case-based Reasoning System for Fault Diagnosis of Aero-engines

Mengqi Chen [a], Rong Qu [b], Weiguo Fang [a,c,*]

[a] School of Economics and Management, Beihang University, Beijing 100083, China;
[b] Computational Optimization and Learning Laboratory, School of Computer Science, University of Nottingham, Nottingham, United Kingdom;
[c] Key Laboratory of Complex System Analysis, Management and Decision (Beihang University), Ministry of Education, Beijing 100083, China;
* Corresponding author: wgfang@buaa.edu.cn

Abstract: Fault diagnosis in aero-engines typically requires an experienced expert for understanding and detecting the cause of faults. However, accurate and quick identification of fault parts is difficult for maintenance crews owing to the complexity of aero-engines. In this study, we developed a case-based reasoning (CBR) system with a highly accurate novel similarity measure for fault diagnosis of aero-engines by retrieving similar fault cases. The proposed CBR system is established based on 143 cases with the knowledge of correctly diagnosed and successfully resolved aero-engine faults, which constitutes the first tentative case base in the field of aero-engine fault diagnosis. The proposed case similarity measure for fault diagnosis of aero-engines integrates three local similarity measures associated with different attributes, especially among which a tree-based semantic similarity measure is proposed to define the relationship between the fault part and fault mode based on a semantic graph incorporated into the aero-engine tree structure. The proposed CBR system is evaluated using the k-nearest neighbors algorithm and 5-fold cross-validation. The system exhibited high retrieval accuracy with all cases collected from real-world scenarios of aero-engine fault diagnosis. Our study shows great promise that the experience-based decisions yielded from the results can aid in aero-engine maintenance and support services.

Keywords: Case-based reasoning; Aero-engines; Fault diagnosis; Textual similarity; Similarity measure.

1. Introduction

Case-based reasoning (CBR) is an experience-based method that solves new problems by retrieving and reusing problem-specific knowledge from similar historical cases or existing experience (Kolodner, 1992). Herein, a new input case, referred to as a target case, is compared with existing cases stored in a historical database called a case base, and the most similar case(s) are retrieved from the case base for reuse (Aamodt & Plaza, 1994). The essence of using previous experience in CBR is similar to that of the human thought process (Cheng & Ma, 2015).

CBR has several advantages over other methods, such as expert systems and knowledge-based systems, which collate expert experience to obtain evaluation rules and object models. This type of knowledge engineering requires exceptional skills and extensive expertise, which bottlenecks the knowledge elicitation during the development of knowledge-based systems (Fyfe & Corchado, 2001). Conversely, CBR is not subjected to modeling bottlenecks because the knowledge is extracted directly from historical cases to be modeled and retained in the case base, which is easier to maintain and update in comparison with that of the rule-based systems (Watson & Marir, 1994). Additionally, when the rules and judgments are not complete in a field, rule-based reasoning methods may fail to return any solution or generate insufficiently accurate solutions. Conversely, CBR always retrieves the most similar historical cases that can solve new problems.

Thus, CBR has been successfully applied to several problem domains owing to the aforementioned advantages. Recent developments in artificial intelligence and data-based methodologies present a promising research direction for addressing large-scale problems of complex fault diagnosis in various fields (Guo, 2020), such as medical (Nasiri, Zahedi, Kuntz & Fathia, 2019) and mechanical engineering (Hu, Qi & Peng, 2015). This can lead to the improvement of existing CBR systems for fault diagnosis of aero-engines without using explicit domain models. The effectiveness of the system can be further improved when more cases are collected in the case base. Moreover, unlike the rule-based expert systems, the case base in a CBR system does not require intensive maintenance. In practice, engineers and maintenance crews can use the CBR system conveniently to make better decisions during outfield tests.

Additionally, CBR is highly suitable for the characteristics of fault diagnosis in aero-engines, which is a complex system comprising different types of parts that render fault diagnosis using model-based methods challenging. In most cases, it is difficult to immediately locate fault parts when an anomaly occurs in aero-engines (Yuan, Wu & Lin, 2016). Additionally, the fault analysis of different types of data in maintenance records warrants expertise in various domains (Cui, et al., 2020). Furthermore, certain information recorded by humans on the faults of outfield flights may be ambiguous and incomplete (Pang, et al., 2020). Moreover, the semi-structured and unstructured fault parts in complex aero-engines render the application of evaluation rules of model-based methods infeasible in aero-engine fault analysis (Tayarani-Bathaie & Khorasani, 2015) (Lu, Huang & Lu, 2017).

In this study, we address several major research issues in the development of a CBR system for fault diagnosis of aero-engines. For instance, identifying, classifying, and structuring the key attributes is

crucial in the representation of fault cases in complex aero-engines. Moreover, the descriptive text must be extracted and collated in the case base to facilitate effective retrieval and maintenance. Finally, the similarity measure between cases must accurately quantify the similarity in text descriptions to retrieve the truly useful cases. Particularly, the actual similarity in terms of different attributes associated with fault diagnosis must be calculated to retrieve truly useful cases. To define the similarity of different fault parts, the irrelevance between parts located at different positions of the aero-engine should be considered to identify the relationship between the fault part and fault mode. Therefore, we propose a novel method to calculate the similarity of fault cases based on the semantic similarity between text descriptions of cases, thus ensuring a quick and accurate fault diagnosis.

The contributions of the study can be summarized as follows.

- First, a novel case similarity measure capable of producing high retrieval accuracy is proposed for the fault diagnosis of aero-engines by integrating three local similarity measures associated with three types of attributes. This novel similarity measure can effectively model and discriminate complex cases in problem domains by considering different types of attributes and semantic similarity in short texts.
- Second, a tree-based semantic similarity measure is proposed to define the relationship between the fault part and fault mode by merging the semantic graph into the endogenous tree structure of the aero-engine itself, thus considering the irrelevance in different fault parts to match the fault mode to the corresponding fault part.
- Third, the 143 cases modeled in the CBR system potentially contributes to the real-life experience-based fault diagnosis of aero-engines in outfield maintenance and support services. Moreover, it provides a complete modeling structure for researchers to study and diagnose aero-engine faults.

The remainder of this paper is organized as follows. Section 2 reviews relevant literature on fault diagnosis of aero-engines, CBR, and textual similarity. Section 3 explains the method for fault diagnosis in aero-engines using three similarity measures based on different types of attributes. The proposed CBR system on aero-engine faults is evaluated and its performance is experimentally verified in Section 4. Finally, Section 5 presents the conclusions of the study.

2. Literature Review

2.1 Fault Diagnosis of Aero-engines

Three main categories of methods, namely the fault tree-based, neural network-based, and mathematical theory-based methods have been investigated in the fault diagnosis of aero-engines.

Fault tree-based methods utilize a logic causality tree to identify the cause of the fault and an ideal approach to reduce risks. Geng, Duan & Li (2011) proposed a dynamic fault tree to support safety analysis of the aircraft considering a higher number of factors for safety modeling. Huang, Wang & Liu, (2012) constructed a novel dynamic fault tree that provided a complete description of the system and applied the method to the aircraft power supply.

Neural network-based methods simulate the human neuron networks to extract and handle unstructured information during information processing. Gou, et al., (2020) applied continuous wavelet transform to convert signal recognition problems to image recognition problems. Additionally, a convolutional neural network was used to identify the features of the image and recognize faults in aero-engine control systems. Furthermore, an autoassociative neural network was used by Li et al., (2020) to diagnose faults in control systems without establishing a model.

Mathematical theory-based methods often require the establishment of an accurate mathematical model. Herein, grey theory, information entropy, and fuzzy mathematics are used for fault diagnosis of aero-engines. A soft-squared pinball-loss function was proposed for training the samples to improve the classification performance of the algorithm and was applied to fault diagnosis of the gas path in aero-engines (Cao, Zhang, Wang & Bai, 2020). Furthermore, Bai, Li, Zhang & Zhao (2020) optimized an immune algorithm and grey theory and proposed a novel mutation strategy to improve the accuracy of fault diagnosis.

2.2 CBR

CBR is an intelligent methodology that solves new problems by retrieving similar historical cases and adapting their solutions, outcomes, and recommendations. Fig. 1 depicts the four sub-phases of a standard CBR cycle, namely retrieve, reuse, revise, and retain (Aamodt & Plaza, 1994).

- Retrieve: To solve a target case, case retrieval is performed as a core process of the CBR cycle, wherein the most similar case(s) are retrieved from a case base. The assessment of the CBR system is significantly affected by the accuracy of case retrieval (Hu, Xia, Skitmore & Chen, 2016) (Liao, Zhang & Mount, 1998). The case retrieval highly relies on four elements, namely the case presentation, case indexing, retrieval algorithms, and similarity measurement (Silva, Carvalho & Caminhas, 2020). The accuracy of retrieval can be improved by constantly adding target cases to extend the case base (Ke, et al., 2020).
- Reuse: If the retrieved case is sufficiently similar to the target case, the solution of the retrieved case can be used directly to solve the target case.
- Revise: However, solutions of the retrieved cases need to be revised in most scenarios and adapted for the target case.

- Retain: After confirming the final solution for the target case, the case and its solution can be retained in the case base using memory mechanisms and dynamic maintenance operations (Yan, Qian & Zhang, 2014).

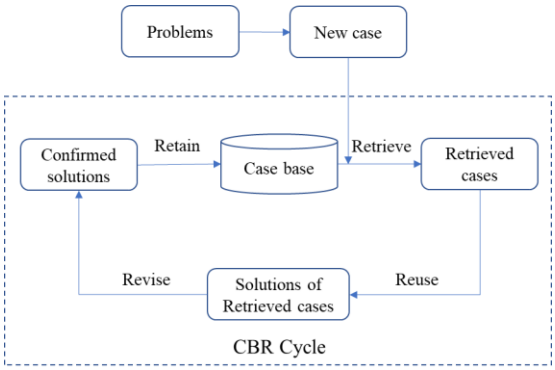

Fig. 1. Case-based reasoning (CBR) process (Aamodt & Plaza, 1994)

Most studies considered the influence of similarity algorithms on the accuracy of case retrieval as it is a key aspect in building a successful and effective CBR. Bannour, Maalel & Ghezala (2020) proposed a two-stage case retrieval method for short texts that included syntactic and cumulative prospect theory-based similarity measurements for crisis response. Similarly, Chang, Lee & Wang, (2016) reported a retrieval method of integrating short-text semantic similarity with recognizing textual entailment for product information retrieval. Based on the development of online knowledge databases, a novel method was proposed by combining Wikipedia with semantic similarity to solve the problem of limited information and sparse features of short texts (Li, Li, Zhang, et al., 2020). Moreover, other researchers investigated the global optimization of CBR. Considering the interactions among attributes, Fei & Feng, (2020) used attitudinal Choquet integral to optimize the global similarity with respect to the importance of attributes. Ahn & Kim (2009) optimized attribute weights with genetic algorithms to devise an effective similarity measure and increased the accuracy of CBR in retrieving the most useful cases.

Considering the other three phases in the CBR cycle, Zhong, Xie & Lin (2015) proposed a two-layer model with a random forest algorithm to improve the accuracy of case reuse. Furthermore, Zhai, Martínez, Martínez & Díaz (2020) developed a method of learning-based adaptation strategy to compare the differences between target cases and previous cases to improve case revision. Yan, Qian & Zhang (2014) and Salamo & Lopez-Sanchez (2011) used an adaptive CBR model to maintain the case base by adding or removing cases. The aforementioned methodologies in the four phases of the CBR cycle contribute to improving the accuracy and effectiveness of the CBR system considering different aspects.

2.3 Textual Similarity Method: Lexical Similarity and Semantic Similarity

The textual similarity method is a basic application of natural language processing to quantify the similarity between different texts, such as phrases, sentences, and documents (Chergui, Begdouri & Groux-Leclet, 2019). A major challenge of applying the textual similarity method is distinguishing the similarity of words or documents in string sequences, wherein two different words may express the same meaning owing to the diversity of descriptions (Do, Roth, et al., 2009). Therefore, the textual similarity method is primarily divided into lexical and semantic similarities based on the object of comparison (Gomaa & Fahmy, 2013).

Lexical similarity computes the similarity between two texts by comparing the distance of string sequences (Gomaa & Fahmy, 2013). Several well-established string-based distance measurement methods, such as Levenshtein distance (Li & Liu, 2007), longest common sequence (Hunt & Szymanski, 1977), and Hamming distance (Apostolico, Guerra, et al., 2016) have been proposed. Another widely used lexical similarity method is the vector space model (VSM), wherein terms are treated as vectors. The commonly used metrics of VSM include cosine similarity (Sidorov, Gelbukh, et al., 2014) and Euclidean distance (Huang, 2008).

Typically, the lexical similarity method is used to calculate the similarity between long texts based on their ambiguity and synonymy (Thiagarajan, Manjunath & Stumptner, 2008). However, as the lexical similarity method cannot capture the semantics of texts, it may lower the accuracy of short texts with identical meanings yet different expressions.

Conversely, semantic similarity computes the similarity between texts by considering semantic features of the meaning represented (Kenter & Rijke, 2015). The two primary categories of semantic similarity include corpus-based and knowledge-based methods (Kadupitiya, Ranathunga & Dias, 2016).

Corpus-based semantic similarity calculates the similarity of texts based on the information from a large corpus, which is a collection of multiple reference texts. A commonly used corpus-based semantic similarity is distributional representation, wherein corpus transforms the text into a vector representation of semantic features. Subsequently, the semantic similarity is calculated based on the similarity of vectors. Dinu & Lapata (2010) represented the meanings of words as a distribution and proposed a framework for characterizing the meanings of words and computing similarity in contexts. Additionally, Ganesh, Kumar & Soman (2016) used distributional representation to classify the texts of health information.

Corpus-based semantic similarity assumes that words of similar meanings have similar contexts (Shi, 2016). Therefore, the calculation of corpus-based semantic similarity usually generally relies on large-scale text data to compare the similarity of input texts and cases stored in a case base.

Conversely, the knowledge-based similarity method quantifies the degree of semantic relevance between texts using information stored in a knowledge base. This method considers the true meaning of words in texts (Mihalcea, Corley & Strapparava, 2006) using a semantic dictionary or semantic network that is organized based on the structural relationship between concepts. Herein, the similarity is computed considering the hypernymy, hyponymy, or synonymy relationships between concepts. The similarity of the two concepts relies on the distance between their corresponding nodes (Resnik, 1995). Schuhmacher & Ponzetto (2014) proposed a semantic model to acquire the information of related entities using the DBpedia knowledge database and calculated the semantic similarity using a graph edit distance-based method. This is similar to the method proposed by Milne & Witte (2008), who developed a Wikipedia link-based measure to compute the semantic similarity using the hyperlink structure of Wikipedia. Chergui, Begdouri & Groux-Leclet (2019) described the knowledge-based similarity method as a graph-based similarity method because it generally uses knowledge or semantic graphs to represent semantic relationships. Moreover, Liu & Xu (2012) divided the graphs into tree-based and directed graph-based methods considering the structure of the knowledge database.

In this study, the cases are extracted from the text descriptions and divided into several keywords, which are short texts with limited context. Therefore, a novel similarity measure of short texts is proposed to characterize the similarity between cases. Additionally, we chose the knowledge-based similarity method to calculate the similarity of fault cases in aero-engines considering the small number of cases in the early stages of case base establishment.

3. Novel Case Similarity Measure for Fault Diagnosis

In this study, when a new input case, defined as a textual description of fault diagnosis, is input to the CBR system, the most similar cases from the case base are retrieved with correct fault classification considering both local and global similarities in the similarity measure. Initially, the local similarity is calculated between the new input case and the retrieved historical cases (Section 3.3). Subsequently, the global similarity between cases is calculated considering all attributes and their corresponding weights (Section 3.4).

3.1 Pre-processing

Typically, fault diagnosis cases include the maintenance date, locations of fault parts, and operation condition of aero-engines. The useful information in the descriptions of engine fault records is manually extracted into several keywords from the fault description. All useful keywords of cases can be divided into 12 attributes, which are used to pre-process the fault diagnosis cases that are transferred and represented as cases in CBR. The useful information is passed to the next step of similarity calculation, whereas the useless information, such as stop-words in the records, are filtered during pre-processing. Fig. 2 illustrates the pre-processing of an input case.

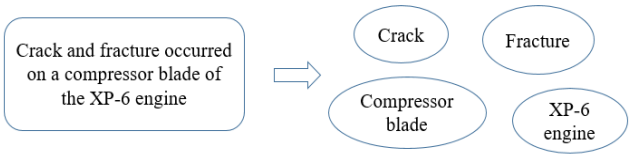

Fig. 2. Pre-processing to extract useful keywords

Based on the extracted keywords with all useful information from the fault descriptions, a fault instance is described using 12 attributes, namely (1) aero-engine model, (2) aero-engine category, (3) aero-engine operation state, (4) thrust performance, (5) temperature performance, (6) rotational performance, (7) aero-engine shutdown, (8) other anomalies, (9) flight height, (10) flight speed, (11) aero-engine fault part, and (12) fault mode. In the similarity calculation, these attributes of the target case are compared with those of the fault diagnosis cases in the case base, which are also processed and represented using these 12 attributes. Additionally, every fault case has its fault positioning and type defined by an expert along with the fault reason and solutions. This information is divided into 5 outputs, including (1) fault positioning, (2) fault type, (3) fault reason, (4) troubleshooting measure, and (5) improvement measure. In this study, if the fault positioning and type of the target case were identical to that of the calculated historical cases, we considered them to be similar. The remaining three outputs were used to confirm the cause and solutions of the fault case to reduce the possibility of faults in the future.

3.2 Case Representation

The 12 attributes are classified into three categories based on their data types, namely the categorical attributes, numerical attributes, and semantic attributes (Table 1). The similarity of attributes in each category is calculated using different similarity methods. The details are explained in Section 3.3.

In the case of categorical attributes, the information is grouped into several categories based on the descriptions of cases. However, considering the different operation temperatures of multiple types of aero-engines, it is not reasonable to compare the service temperatures of different aero-engines. For instance, an exhaust gas temperature of 850 °C is considered normal for XP-6 aero-engines, whereas it causes overheat in XP-7 aero-engines. Therefore, temperature conditions of different categories are concerted based on the handbook of each type of aero-engine.

In the case of numerical attributes, the information is represented by numerical values.

Table 1 Classification of attributes

Classifications	Attributes	Description
Categorical attributes	Aero-engine model	Identify specific engine models, including XP-7, XP-8, and other models
	Aero-engine category	Identify the category of engine usage, i.e., military or civilian
	Aero-engine operation state	Identify the working state of the engine in case of faults, including intermediate, maximum, and other states
	Thrust performance	Identify symptoms of abnormal thrust in engines, including thrust loss, thrust fluctuation, and other factors
	Temperature performance	Identify symptoms of abnormal temperature in engines, including overheat, temperature drop, and other factors
	Rotational performance	Identify symptoms of abnormal rotational speed in engines, including speed-drop, speed fluctuation, and other factors
	Aero-engine shutdown	Identify whether the engine stops in the air (Yes or No)
	Other anomalies	Identify other engine anomalies, including engine surge, abnormal engine sound, and other anomalies
Numerical attributes	Flight height	Identify the flight height of aircraft in case of engine fault, measured in kilometers
	Flight speed	Identify the flight speed of aircraft in case of engine fault, measured in Mach number
Semantic attributes	Aero-engine fault part	Identify the semantic terms of specific fault parts, such as compressor blade, turbine blade, and other parts
	Fault mode	Identify the semantic terms of fault mode corresponding to specific fault parts, such as fracture, crack, and other faults

In semantic attributes, the complex semantic relationship between fault part and fault mode is considered owing to the diverse relationship of each part.

Table 2 summarizes three examples of case presentation. All cases in a case base are presented in this structured manner. The five outputs of the cases include (1) fault positioning, which indicates a certain system, a specific structure, or a component in aero-engine; (2) fault type, which specifies the feature that a fault exhibits. Every fault type is associated with a case set in which all cases are of the same fault type; (3) fault reason, which explains the reason for a fault case; (4) troubleshooting measure, which indicates the methods of troubleshooting a fault case; and (5) improvement measure, which identifies the methods of reducing faults and improving product reliability in the future.

Table 2 Case presentation of three example cases

Case Description	On 4th March 1978, a military engine of type XP-6 was tested at a height of 2000 m. When the pilot turned the engine to its maximum thrust state, a loud noise was heard and the rotating speed decreased. When the pilot throttled back, the engine shut down.	In March 1990, when the pilot pushed the throttle to test a military engine of type XP-7, the engine shutdown in-flight at a flight height of 9.5 km and speed of 0.6 Mach.	When an aircraft equipped with the XP-7 engine climbed sharply to a height of 20 km, the thrust of the engine decreased, and the exhaust gas temperature (EGT) increased higher than the prescriptive temperature of 820 °C.
Aero-engine model	XP-6	XP-7	XP-7
Aero-engine category	Military	Military	Military

Aero-engine operation state	Maximum power state	Intermediate state	Intermediate state
Flight height (Km)	2	9.5	20
Thrust performance	Normal	Normal	Engine thrust decline
Temperature performance	Normal	Normal	Over-temperature (T4)
Flight speed (Ma)	/	0.6	/
Aero-engine shutdown	Yes	Yes	No
Rotational performance	Speed-drop	Normal	Normal
Other anomalies	Abnormal engine sound	Engine surge	/
Aero-engine fault part	Sleeve of fuel pump	/	/
Fault mode	loose	/	/

Table 3 presents an example of the five outputs. Among these, fault positioning and fault type are used to classify the cases and the other three outputs represent the reasons and solutions of cases.

Table 3 An instance of five outputs

Case No.	Fault positioning	Fault type	Fault reason	Troubleshooting measure	Improvement measure
9	Anomaly of engine performance	In-flight shutdown (IFSD)	Improper matching between inlet and aero-engine; Pilot mis-operation; Machining error of parts	Changing the inlet cone from three-step adjustment to infinite adjustment; Modifying the airfoil of compressor blades	Using a new main fuel pump regulator; Adding engine surge prevention system

3.3 Local Similarity Measure

3.3.1 Categorical Attributes

In the CBR system, categorical attributes are classified into the aero-engine model, aero-engine category, aero-engine operation state, thrust performance, temperature performance, rotational performance, aero-engine shutdown, and other anomalies. The similarity between categorical attributes is calculated using Eq. (1).

$$sim_{a(a_j^{nc},a_j^{c_i})} = \begin{cases} 0 & if \quad a_j^{nc} \neq a_j^{c_i} \\ 1 & if \quad a_j^{nc} = a_j^{c_i} \end{cases} \tag{1}$$

where a_j^{nc} and $a_j^{c_i}$ denote the values of attribute j in the new input case and case i in the case base, respectively. Further, $i \in \{1,2,...,I\}$, $j \in \{1,2,...,J\}$, where I and J indicate the total number of attributes and cases, respectively.

3.3.2 Numerical Attributes

Based on the collected fault cases of aero-engines, numerical attributes are categorized into flight height and flight speed. The similarity is calculated as the normalized distance between the two values of numerical attributes using Eq. (2) and (3).

$$sim_{b(b_j^{nc},b_j^{c_i})} = 1 - DIST(b_j^{nc}, b_j^{c_i}) \tag{2}$$

$$DIST(b_j^{nc}, b_j^{c_i}) = \frac{\left| b_j^{nc} - b_j^{c_i} \right|}{b_j^{\max} - b_j^{\min}} \tag{3}$$

Where b_j^{nc} and $b_j^{c_i}$ denote the values of attribute j in the new input case and case i in the case base, respectively. Further, $b_j^{\max}$ and $b_j^{\min}$ represent the maximum and minimum values of attribute j, respectively.

3.3.3 Tree-based Semantic Similarity

The tree-based semantic similarity calculates the semantic similarity of the fault part and fault mode in aero-engines. In this study, we considered whether it is reasonable to calculate the similarity between different fault parts with the same fault mode.

As aero-engines comprise numerous parts, the faults of different parts are generally irrelevant. For instance, the crack on the compressor blade is typically caused by the high cycle fatigue, whereas a crack on a turbine blade is primarily caused by thermal–mechanical fatigue. Consequently, the failure mechanism for the same fault mode of "crack" in both compressor and turbine blades is entirely different. Therefore, it is unreasonable to compare the crack on the compressor blade with that on the turbine blade. During modeling, the comparison of these modes in CBR can reduce the efficiency of the retrieval if all cases in the case base are checked against the target case, which in turn increases the computational cost with the extension of the case base. In the proposed CBR system, only the fault of the same part in different cases is considered owing to the irrelevance of certain parts in the aero-engine.

Therefore, the key factor in this study was to define the relationship between an aero-engine fault part and the corresponding fault mode of every case and model it using a semantic similarity measure. Thus, the tree-based semantic similarity is proposed to define the relationship between the fault mode and the corresponding fault parts. It comprises two sections, namely the fault part tree structure and semantic graph of fault mode.

Fig. 3 depicts a partial tree structure of fault parts of an aero-engine, wherein different parts are structured and distinguished to calculate their similarities with the same or different fault mode. Several parts share certain similarities within the entire aero-engine structure despite every part being irrelevant from each other in aviation engineering. The tree structure quantifies the similarity of different parts, which relies on the number of their parent nodes.

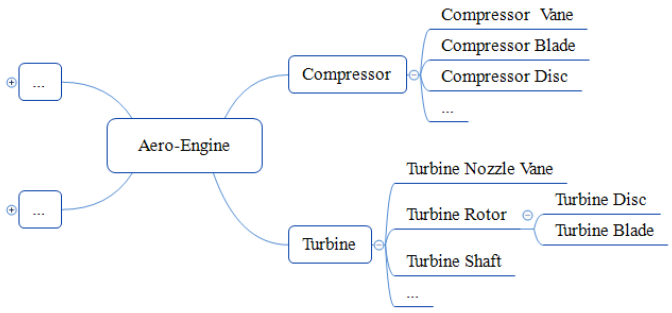

Fig. 3. Fault part tree structure of an aero-engine

Furthermore, we developed the semantic graph of fault mode based on online knowledge databases, including Wikipedia and DBpedia, which contain billions of triples reflecting the semantics using a dense link structure and have been widely adopted in several studies previously (Li, Li, et al., 2020) (Chergui, Begdouri & Groux-Leclet, 2019) (Begdouri, Chergui & Leclet-Groux, 2018). The missing semantic information of aero-engine fault mode in Wikipedia has been continuously added by experts or engineers with their extensive knowledge in aero-engines. Fig. 4 illustrates a fault mode of aero-engines.

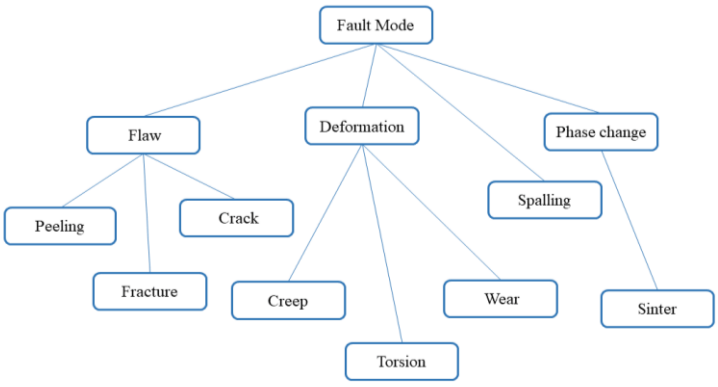

Fig. 4. Fault mode of aero-engines

As depicted in Fig. 5, the tree structure and semantic graph are combined to define the relationship between each fault part and the corresponding fault mode in the proposed CBR system.

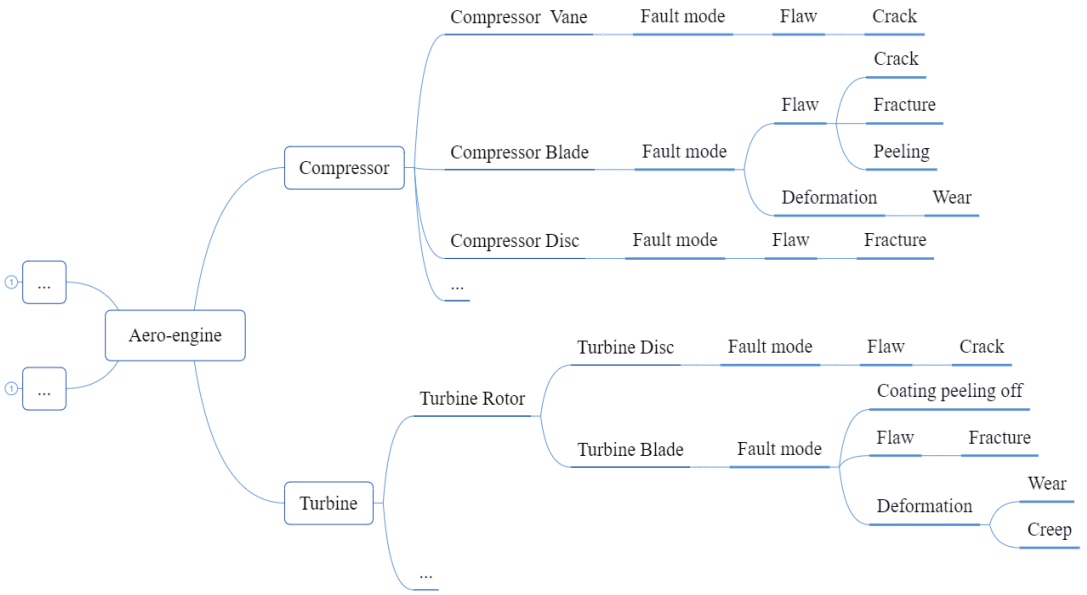

Fig. 5. Tree-based semantic graph for aero-engine faults

The tree-based semantic similarity is based on the following three assumptions.

(1) Two different nodes with a shorter distance within the tree-based semantic graph are more similar than those with a longer distance. The shortest path between n_i and n_j can be denoted as *Dist* (n_i, n_j). For example, *Dist ("Crack (Compressor vane)", "Fracture (Compressor Blade)") = 8*, and *Dist ("Crack (Compressor Blade)", "Peeling (Compressor Blade)") = 2*.
Thus, *Crack (Compressor Blade)* is more relevant to *Peeling (Compressor Blade)* than to *Crack (Compressor vane)*.

(2) The correlation of two nodes increases with the increase in the distance between the nearest shared parent node and root node. In other words, the correlation of two nodes can be considered as the depth of their nearest shared parent node. When the taxonomy depth of the two nodes is high and the nearest shared parent node is far from the root node, it indicates that the nodes are more similar in semantics.

The nearest shared parent node n_k between nodes n_i and n_j can be denoted as *Nspn* (n_i, n_j). For example, *Nspn ("Wear (Turbine Blade)", "Creep (Turbine Blade)") = Deformation (Turbine Blade)*, which implies that *Deformation (Turbine Blade)* is the nearest shared parent node between *Wear (Turbine Blade)* and *Creep (Turbine Blade)*. Furthermore, *Nspn ("Wear (Turbine Blade)", "Crack (Turbine Disc)") = Turbine Rotor* indicates that *Turbine Rotor* is the nearest shared parent node between *Wear (Turbine Blade)* and *Crack (Turbine Disc)*.

(3) The depth, denoted as *Depth*(n_k), represents the distance from a node to the root node. For example, *Dept (Turbine Rotor) = 2* and *Dept (Deformation (Turbine Blade)) = 5* imply that the depth of *Turbine Rotor* and *Deformation (Turbine Blade)* is 2 and 5, respectively.

The tree-based semantic similarity can be calculated using Eq. (4) based on the three aforementioned assumptions. It is defined as a function of the location of the node in taxonomy (Meng, Huang, Gu, 2013) (Wu & Palmer,1994).

$$Sim_c(c_j^{nc}, c_j^{c_i}) = \frac{2 \times Depth(Nspn(c_j^{nc}, c_j^{c_i}))}{Dist(c_j^{nc}, c_j^{c_i}) + 2 \times Depth(Nspn(c_j^{nc}, c_j^{c_i}))} \quad (4)$$

where c_j^{nc} and $c_j^{c_i}$ denote the values of attribute j in the new input case and case i in the case base, respectively.

3.4 Global Similarity Measure

The global similarity between cases calculates the weighted sum of the similarities with different attributes by integrating their local similarities, as indicated in Eq. (5)

$$Sim(nc, c_i) = \sum_{j=1}^{J} w_j \times Sim_k(k_j^{nc}, k_j^{n_i}) \quad (5)$$

where w_j denotes the weight of attribute j, $Sim(nc, c_i)$ represents the global similarity between a target case and a historical case i, and $Sim_k(k_j^{nc}, k_j^{n_i})$ indicates the local similarity considering the three different types of similarity calculations with $k \in \{a, b, c\}$, as explained in Section 3.3. In this study, the weights of all attributes were assumed to be identical.

4. Experimental Validation and Discussion

4.1 Building the Case Base

In this study, the case base comprises the following three components.

(1) A total of 143 cases are collected in the case base and processed using the existing aircraft fault records in Song, Chen, et al. (1993), which provides numerous fault cases in different systems, structures, and components of aero-engines. All cases are stored in the case base consistent with the case structure described in Section 3.2. Every case comprises a detailed description of the fault model, reason, and conclusion. Moreover, the information is divided into different fault positioning and fault types based on the opinions of domain experts. The presented information serves as a reference in our experimental evaluations detailed in Section 4.2.

(2) Each case is stored in the case base with 12 attributes and 5 outputs as explained in Section 3.1.

(3) A semantic graph characterizes different fault modes in the aero-engine with 24 nodes, which contain all fault modes in the case base. Fig. 4 depicts a part of the semantic graph of fault mode in aero-engines. Furthermore, the tree-based semantic graph is adopted to model the relationship between parts and the corresponding fault mode (Fig. 5). We considered 54 fault parts in this study, wherein each fault part has one to three fault modes located in the semantic graph.

4.2 Experimental Validation Methodology

Two methodologies, namely the 5-fold cross-validation and *k*-nearest neighbors, were used to test the performance of the CBR system with a partition of data to evaluate the retrieval accuracy. The accuracy of the CBR system was determined using the fault positioning and fault type of the most similar cases retrieved from the case base. The retrieved results were sorted according to their similarity to the target case. The fault positioning and fault type of every case in the proposed case base are defined by experts. The most similar cases are those with both the highest similarity and identical fault positioning and fault type as those of the target case.

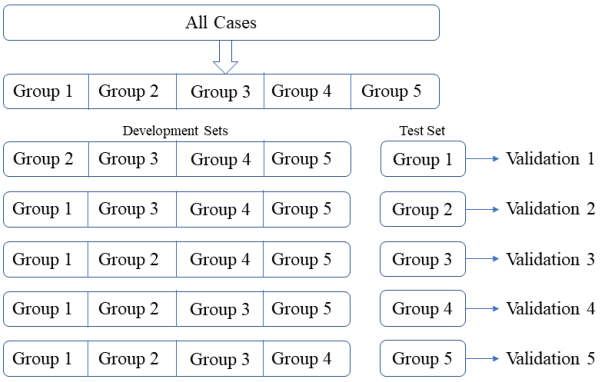

Fig. 6. Implementation of the 5-fold cross-validation

The 5-fold cross-validation (Kohavi, 1995) evaluated the performance of the CBR system, wherein the 143 cases were divided into five partitions with an approximately equal size based on their classifications. Groups 1, 2, and 3 contain 29 cases each, whereas Groups 4 and 5 contain 28 cases each. Four among the five groups were considered as the development set and the remaining group was dynamically considered as the test set. As depicted in Fig. 6, the validation operation is repeated five times, wherein each validation considers one partition as the test set and the other four as the development set.

The cross-validation implemented in this study differs from regular cross-validations, wherein the entire dataset is divided into training sets and test sets. Typically, the training set is used to train the parameters in modeling the explicit or implicit relationship between input and output variables, whereas the test set is used to verify the model. An advantage of the CBR system is that it does not require the establishment of an explicit or implicit relationship between the input and output variables. The development set in this study indicates the collection of correctly diagnosed and successfully resolved aero-engine fault cases, which serve as benchmarks for evaluating new fault diagnosis and resolutions.

The performance of the proposed CBR system is calculated as the ratio of the number of test cases that are correctly matched to the total number of test cases, as indicated in Eq. (6).

$$Accuracy = \frac{Number\ of\ test\ cases\ correctly\ matched}{Total\ number\ of\ test\ cases} \tag{6}$$

The *k*-nearest neighbors (KNN) algorithm is used to determine whether the retrieved cases are correctly matched to each test case. The classification of a sample is determined by the classification of KNN. For each test case in the proposed CBR system, the test target case is considered to be correctly matched if the classification pertaining to most cases in the retrieved *k* most similar cases is identical to that of the target case.

4.3 Results and Discussion

Table 4 summarizes the results of one example of Validation 1 with a test set of Partition 1. For each case in the test set, the top 10 most similar cases in the development set of 29 cases are retrieved and sorted as presented in the table. For instance, the values of 0.8(2) on the second row and second column imply that Case No. 2 in the development set has the highest similarity of 0.8 to Case No. 1 in the test set.

Table 4 Test result of Partition 1 in Validation 1

Case No.	Top 1	Top2	Top3	Top4	Top5	Top6	Top7	Top8	Top9	Top10
1	0.8 (2)	0.57 (3)	0.33 (16)	0.33 (17)	0.25 (5)	0.25 (6)	0.16 (136)	0.16 (138)	0.14 (8)	0.14 (9)
4	1 (5)	1 (6)	0.56 (134)	0.5 (8)	0.5 (9)	0.38(11)	0.38(12)	0.38(13)	0.38 (14)	0.38 (78)
7	1 (8)	1(9)	0.57(11)	0.57(12)	0.57(13)	0.57(14)	0.5 (5)	0.5 (6)	0.5 (108)	0.5 (109)
18	1 (19)	0.90 (24)	0.90 (25)	0.81 (21)	0.81 (22)	0.81 (23)	0.81 (26)	0.81 (28)	0.56 (56)	0.56 (57)
20	1 (22)	1 (23)	0.90 (21)	0.81 (25)	0.81 (19)	0.81 (21)	0.81 (22)	0.81 (23)	0.53(29)	0.36 (105)
24	1 (25)	0.90 (19)	0.81 (21)	0.81 (22)	0.81 (23)	0.81 (26)	0.81 (28)	0.44(50)	0.38(51)	0.38 (53)
27	1 (26)	1 (28)	1 (29)	1 (30)	1 (31)	1 (32)	0.90 (21)	0.81 (19)	0.81 (20)	0.44 (50)
35	0.97 (37)	0.97 (36)	0.81 (26)	0.81 (28)	0.81 (29)	0.81 (30)	0.81 (31)	0.81 (32)	0.76 (33)	0.26 (113)
40	0.95 (43)	0.93 (41)	0.93 (42)	0.29 (60)	0.27 (66)	0.27 (67)	0.27 (69)	0.27 (70)	0.23 (59)	0.13 (17)
46	0.97 (47)	0.97 (48)	0.97 (49)	0.63 (51)	0.63 (44)	0.61 (45)	0.61 (53)	0.61 (54)	0.6 (50)	0.58 (56)
52	1 (51)	1 (53)	1 (54)	0.94 (61)	0.94 (62)	0.94 (63)	0.94 (64)	0.94 (65)	0.93 (50)	0.39 (113)
55	0.96 (56)	0.94 (58)	0.94 (57)	0.64 (47)	0.64 (48)	0.64 (49)	0.64 (53)	0.64 (54)	0.63 (44)	0.63 (45)
61	1 (62)	1 (63)	1 (64)	1 (65)	0.94 (53)	0.94 (54)	0.44 (47)	0.44 (48)	0.29 (19)	0.29 (24)
68	0.96 (69)	0.96 (70)	0.96 (71)	0.27 (66)	0.23 (67)	0.09 (73)	0.09 (74)	0.09 (75)	0.09 (77)	0.09 (78)
72	1 (73)	1 (74)	1 (75)	0.67 (82)	0.67 (83)	0.67 (84)	0.67(85)	0.67 (86)	0.36 (125)	0.29 (126)
76	1 (77)	1 (78)	1 (79)	1 (80)	0.83 (82)	0.83 (83)	0.83 (84)	0.83 (85)	0.83 (86)	0.48 (47)
81	1 (82)	1 (83)	1 (84)	1 (85)	1 (86)	0.83 (77)	0.83 (78)	0.83 (79)	0.83 (80)	0.48 (47)
89	0.6 (90)	0.58 (44)	0.58 (45)	0.33 (47)	0.33 (48)	0.33 (49)	0.33 (50)	0.33 (51)	0.33 (53)	0.33 (54)
91	0.6 (92)	0.6 (93)	0.6 (94)	0.46 (73)	0.46 (74)	0.46 (75)	0.46 (87)	0.46 (88)	0.26 (106)	0.14 (8)
99	1 (95)	1 (96)	1 (97)	1 (98)	1 (100)	0.9 (101)	0.44 (105)	0.39 (104)	0.14 (66)	0.14(67)
102	0.83 (104)	0.58 (105)	0.58 (106)	0.58 (107)	0.58 (108)	0.58 (109)	0.58 (110)	0.58 (111)	0.47 (5)	0.47 (6)
103	0.72 (104)	0.53 (105)	0.53 (106)	0.53 (107)	0.53 (108)	0.53 (109)	0.53 (110)	0.53 (111)	0.38 (5)	0.38 (6)
112	1 (113)	0.68 (114)	0.68 (115)	0.49 (117)	0.49 (118)	0.48 (120)	0.48 (121)	0.48 (122)	0.48 (123)	0.23(59)
116	0.83 (117)	0.83 (118)	0.4 (113)	0.35 (114)	0.35 (115)	0.33 (135)	0.29 (136)	0.29 (138)	0.29 (141)	0.14 (8)
119	0.5 (120)	0.5 (121)	0.5 (122)	0.5 (123)	0.4 (114)	0.4 (115)	0.38 (130)	0.38 (131)	0.38 (132)	0.16 (136)
124	0.79 (125)	0.47 (126)	0.38 (113)	0.38 (120)	0.38 (121)	0.38 (122)	0.38 (123)	0.29 (114)	0.29 (115)	0.13 (51)
127	1 (128)	0.36 (125)	0.29 (126)	0.29 (130)	0.29 (131)	0.29 (132)	0.2 (135)	0.2 (142)	0.2 (143)	0.12 (2)
129	0.8 (130)	0.8 (131)	0.8 (132)	0.46 (114)	0.46 (115)	0.38 (119)	0.36 (5)	0.36 (6)	0.36 (8)	0.36 (9)
133	0.75 (134)	0.5 (5)	0.5 (6)	0.33 (16)	0.33 (17)	0.33 (140)	0.25 (141)	0.14 (136)	0.14 (138)	0.09 (2)

The parameter *k* in the KNN algorithm is varied from 1 to 10 to observe the performance of the CBR system. The accuracy varies with different values of *k*. On average, the number of cases under each fault type is small as the number of cases in the case base is only 143. Therefore, a relatively small value of *k* is selected to evaluate the accuracy of the CBR system. Table 5 presents the accuracy of Validation 1 with different values of *k*.

Table 5 Accuracy of Validation 1 with different values of *k*

k	1	2	3	4	5	6	7	8	9	10
Accuracy	0.9655	0.8966	0.8966	0.6552	0.5862	0.4828	0.4828	0.4483	0.4138	0.4138

Fig. 7 illustrates the retrieval accuracy of five validations and the average accuracy with respect to different values of *k*. We observed that when $k = 1–3$, the performance of CBR is adequate with more than 80% accuracy. The highest accuracy of 0.958 is obtained at $k = 1$, which implies that 95.8% of the returned cases with the same fault positioning and fault type are similar.

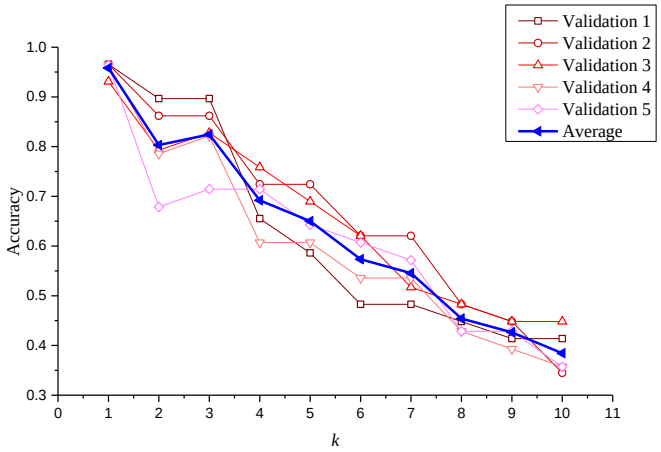

Fig. 7. Retrieval accuracy with different values of *k*

It is worth mentioning that based on the classification of KNN, the retrieval accuracy decreases with *k*. When the value of *k* is sufficiently large, KNN is rendered meaningless as all cases exist in the same classification. The case base in this study comprises 143 cases and 38 fault types. This implies that for every fault type, only a small number of cases exist on average. Additionally, only one or two relevant cases exist for certain fault types owing to the extremely severe consequences of the fault. Therefore, the accuracy is the highest at $k = 1$ and extremely low when *k* is large. However, this does not affect the performance of the CBR system because the aim of the 5-fold cross-validation and KNN is to identify the most suitable value of *k* for CBR to attain the highest accuracy. In the future, the case base can be extended by adding new fault cases of aero-engines. The similarity measure can be further improved with a higher number of cases in the case base and by fine-tuning the attribute weights.

5. Conclusions

In this study, we developed a CBR system to ease the decision-making in aero-engine fault diagnosis. A novel case representation and similarity measure are proposed for fault diagnosis of complex aero-engines. We established an aero-engine fault case base with 143 cases. The keywords are extracted from a linguistic description and divided into three types of attributes, namely categorical, numerical, and semantic attributes. Additionally, three similarity measures are proposed based on different attributes in these categories. Particularly, a semantic correlation between the fault part and its corresponding fault mode is considered, and a tree structure combined with a semantic graph-based approach is devised to quantify the similarity between semantic attributes. The most similar case is retrieved based on the combined local and global similarities. The KNN algorithm and 5-fold cross-validation are used to evaluate the performance of the CBR system by partitioning all cases into a development set and a test set. The experimental results exhibited a high accuracy in terms of retrieving similar cases for target cases.

The accurate diagnosis and resolution of aero-engine faults have always been an important issue in the field of aviation maintenance and support service. Despite the extensive knowledge of aero-engine fault diagnosis, a higher number of cases need to be accumulated and modeled to develop and utilize resources and assist decision-making in aero-engine fault diagnosis. This study forms the basis for the development of an effective CBR system with an initial set of actual cases, which presents a promising high accuracy of case retrieval.

The era of big data has increased the convenience of the collection and collation of aero-engine fault cases. However, the data on fault cases are generally semi-structured or unstructured text descriptions, rendering it highly challenging to develop and maintain quantitative models or rule-based models that can capture the causal relationships of faults. CBR is an experience-based methodology that does not require explicit models, thus addressing complex problems, such as the one considered in this study. Apart from avoiding the bottleneck issue of modeling, the case base in the CBR system can be easily maintained, updated, and extended with new successfully resolved cases, thus improving the service support system with higher accuracy of case retrieval.

Similarity measure, which significantly relies on the attributes of interest and their weights, is critical to improving the performance of CBR systems. In the future, we intend to improve the CBR system by incorporating a higher number of potential key attributes that are interactively extracted from maintenance and diagnosis records based on further adjusted weights of attributes. Furthermore, we intend to extend the case base with new cases when available.

References

Kolodner, J.L. (1992). An introduction to case-based reasoning. *Artificial Intelligence Review*, 6 (1), 3-34.

Aamodt, A., & Plaza, E. (1994). Case-based reasoning: Foundational issues, methodological variations, and system approaches. *AI Communications*, 7 (1), 39–59.

Cheng, J.C.P., & Ma, L.J. (2015). A non-linear case-based reasoning approach for retrieval of similar cases and selection of target credits in LEED projects. *Building and Environment*, 93, 349-361.

Fyfe, C., & Corchado, J.M. (2001). Automating the construction of CBR systems using Kernel methods. *International Journal of Intelligent Systems*, 16 (4), 571-586.

Watson, I., & Marir, F. (1994). Case-based reasoning: A review. *Knowledge Engineering Review*, 9(4), 327-354.

Guo, L.F. (2020). Research on civil aviation engine fault diagnosis based on case reasoni-ng. Civil Aviation University of China.

Nasiri, S., Zahedi, G., Kuntz, S., & Fathia, M. (2019). Knowledge representation and management based on an ontological CBR system for dementia caregiving. *Neurocomputing*, 350, 181-194.

Hu, J., Qi, J., & Peng, Y.H. (2015). New CBR adaptation method combining with problem-solution relational analysis for mechanical design. *Computers in Industry*, 66, 41-51.

Yuan, M., Wu, Y.T., & Lin, L. (2016). Fault diagnosis and remaining useful life estimation of aero engine using LSTM neural network. 2016 IEEE/ CSAA International Conference on Aircraft Utility Systems (AUS). 135-140.

Cui, J.G., Tian, Y., Cui, X., et al., (2020). An effective fault diagnosis method for aero e-ngines based on GSA-SAE. *Transactions of Nanjing University of Aeronautics & Astr-onautics*, 37(5), 750-757.

Pang, M.Y., Suo, Z.Y., Zheng, W.Z., et al., (2020). A small sample fault diagnosis of aero engine based on RS-CART decision tree. *Journal of Aerospace Power*, 35(7), 1559-1568.

Tayarani-Bathaie, S.S., & Khorasani, K. (2015). Fault detection and isolation of gas turbine engines using a bank of neural networks. *Journal of Process Control*, 36, 22-41.

Lu, J.J., Huang, J.Q., & Lu, F. (2017). Sensor fault diagnosis for aero engine based on online sequential extreme learning machine with memory principle. *Energies*, 10(1), 39.

Geng, Q.C., Duan, H.B., & Li,S.T. (2011). Dynamic fault tree analysis approach to safety analysis of civil aircraft. 2011 6th IEEE Conference on Industrial Electronics and Applications. 1443-1448.

Huang, Z.T., Wang, Z.S., & Liu, Z.B. (2012). Fault diagnosis of aircraft power supply based on priority dynamic fault tree. *Advanced Materials Research*, 443-444, 229-236.

Gou, L.F., Li, H.H., Zheng, H., Li, H., & Pei, X. (2020). Aeroengine control system sensor fault diagnosis based on CWT and CNN. *Mathematical Problems in Engineering*, 2020, 1-12.

Li, H.H., Gou, L.F., Li, H.C., Xing, X.H., & Yang, J. (2020). Multiple fault diagnosis of aeroengine control system based on autoassociative neural network. 2020 11th Interna-tional Conference on Mechanical and Aerospace Engineering (ICMAE), 107-113

Cao, Y.Y., Zhang, B., Wang, H.W., & Bai, Y. (2020). Gas path fault diagnosis of aeroeng-ine based on soft square pinball loss ELM. *IEEE Access*, 8, 131032-131046.

Bai, Y., Li, Y.J., Zhang, B., & Zhao, Y.C. (2020). Intelligent fault diagnosis of aeroengine based on algorithm fusion. 2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC). 1, 255-261.

Hu, X., Xia, B., Skitmore, M., & Chen, Q. (2016). The application of case-based reasoning in construction management research: An overview. *Automation in Construction*, 72, 65-74.

Liao, T.W., Zhang, Z.M., & Mount, C.R. (1998). Similarity Measures for Retrieval in Cas-e-Based Reasoning Systems. *Applied Artificial Intelligence*, 12, 267-288.

Silva, G.C., Carvalho, E.E.O., & Caminhas, W.M. (2020). An artificial immune systems a-pproach to case-based reasoning applied to fault detection and diagnosis. *Expert Syste-ms with Application*, 140, 112906.1-112906.15.

Ke, C., Jiang, Z.G., Zhang, H., Wang,Y., & Zhu, S. (2020). An intelligent design for remanufacturing method based on vector space model and case-based reasoning. *Jour-nal of Cleaner Production*, 277, 123269.

Yan, A.J., Qian, L.M., & Zhang, C.X. (2014). Memory and forgetting: An improved dynamic maintenance method for case-based reasoning. *Information Sciences*, 287, 50-60.

Bannour, W., Maalel, A., & Ghezala, H.H.B. (2020). Case-based reasoning for crisis response: case representation and case retrieval. *Procedia Computer Science*, 176, 1063-1072.

Chang, J.W., Lee, M.C., & Wang, T.I. (2016). Integrating a semantic-based retrieval agent into case-based reasoning systems: A case study of an online bookstore. *Computers in Industry*, 78, 29-42.

Li, P., Li, T.C., Zhang, S.Z., et al. (2020). A semi-explicit short text retrieval method combining Wikipedia features. *Engineering Applications of Artificial Intelligence*, 94, 103809.

Fei, L.G., & Feng, Y.Q. (2020). A novel retrieval strategy for case-based reasoning based on attitudinal Choquet integral. *Engineering Applications of Artificial Intelligence*, 94, 103791.

Ahn, H., & Kim, K.J. (2009). Global optimization of case-based reasoning for breast cyto-logy diagnosis. *Expert Systems with Applications*, 36 (1), 724-734.

Zhong, S.S., Xie, X.L., & Lin, L. (2015). Two-layer random forests model for case reuse in case-based reasoning. *Expert Systems with Applications*, 42 (24), 9412-9425.

Zhai, Z.Y., Martínez, J.F., Martínez, N.L., & Díaz, V.H. (2020). Applying case-based reas-oning and a learning-based adaptation strategy to irrigation scheduling in grape farmi-ng. *Computers and Electronics in Agriculture*, 178, 105741.

Salamo, M., & Lopez-Sanchez, M. (2011). Adaptive case-based reasoning using retention and forgetting strategies. *Knowledge-Based Systems*, 24 (2), 230-247.

Chergui, O., Begdouri, A., & Groux-Leclet, D. (2019). Integrating a Bayesian semant-ic similarity approach into CBR for knowledge reuse in community question answeri-ng. *Knowledge-Based Systems*, 185, 104919.

Do, Q., Roth, D., Sammons, M., Tu, Y.C., & Vydiswaran, V.G.V. (2009). Robust, light-weight approaches to compute lexical similarity. Computer Science Research and Technical Reports, University of Illinois. (2009).

Gomaa, W.H., & Fahmy, A.A. (2013). A survey of text similarity approaches. *Internation-al Journal of Computer Applications*, 68 (13), 13-18.

Li, Y.J., & Liu, B. (2007). A normalized Levenshtein distance metric. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29 (6), 1091-1095.

Hunt, J.W., & Szymanski, T.G. (1977). A fast algorithm for computing longest common s-ubsequences. *Communications of the ACM*, 20 (5), 350-353.
Apostolico, A., Guerra, C., M. Landau, G., & Pizzi, C. (2016). Sequence similarity measures based on bounded hamming distance. *Theoretical Computer Science*, 638, 76-90.
Sidorov, G., Gelbukh, A., Gómez-Adorno, H., & Pinto, D. (2014). Soft similarity and soft cosine measure: Similarity of features in vector space model. *Computación y Sistemas*, 18 (3), 491-504.
Huang, A. (2008). Similarity measures for text document clustering. *Proceedings of the Sixth New Zealand Computer Science Research Student Conference*, Christchurch, New Zealand. 49-56.
Thiagarajan, R., Manjunath, G., & Stumptner, M. (2008). Computing semantic similarity using ontologies. *the International Semantic Web Conference*, Karlsruhe, Germany.
Kenter, T., & Rijke, M.D. (2015). Short text similarity with word embeddings. *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, 1411-1420.
Kadupitiya, J., Ranathunga, S., & Dias, G. (2016). Sinhala short sentence similarity calculation using corpus-based and knowledge-based similarity measures. *Proceedings of the 6th Workshop on South and Southeast Asian Natural Language Processing*, 44-53.
Dinu, G., & Lapata, M. (2010). Measuring distributional similarity in context. *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*. 1162-1172.
Ganesh, H.B.B., Kumar, M.A., & Soman, K.P. (2016). Distributional Semantic Representation in Health Care Text Classification. 1737, 201-204.
Shi, K.L. (2016). Research and implementation of semantic similarity computing by c-ombining knowledge-based and corpus-based methods. Beijing Jiaotong University.
Mihalcea, R., Corley, C., & Strapparava, C. (2006). Corpus-based and knowledge-based measures of text semantic similarity. *American Association for Artificial Intelligence (Aaai)*, 775-780.
Resnik, P. (1995). Using information content to evaluate semantic similarity in a taxonomy. *Proceedings of the 14th International Joint Conference on Artificial Intelligence (IJCAI)*, 448-453.
Schuhmacher, M., & Ponzetto, S.P. (2014). Knowledge-based graph document modeling. *Proceedings of the 7th ACM International Conference on Web Search and Data Mining*, 543-552.
Milne, D., & Witten, I.H. (2008). An effective, low-cost measure of semantic relatedness obtained from Wikipedia links. *In Proceeding of AAAI Workshop on Wikipedia and Artificial Intelligence*, 25-30.
Liu, H.Z., & Xu, D. (2012). Ontology based semantic similarity and relatedness measures review. *Computer Science*, 39 (2), 8-13.
Begdouri, A., Chergui, O., & Leclet-Groux, D. (2018). A knowledge-based approach for keywords modeling into a semantic graph. *International Journal of Information Science and Technology*, 2 (1), 12-24.
Meng, L., Huang, R., Gu, J. (2013). A review of semantic similarity measures in wo-rdnet. *International Journal of Hybrid Information Technology*, 6 (1), 1-12.
Wu, Z.B., & Palmer, M. (1994). Verb semantics and lexical selection. *In: Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics*, New Mexico, USA. 133–138.
Song, Z.H., Chen, G. et al. (1993). Analysis of Typical Failures of Aeroengine, Beijing University of Aeronautics and Astronautics Press. (In Chinese).
Kohavi, R. (1995). The power of decision tables. *In Proceedings of the 8th European Conference on Machine Learning (ECML'95)*, Springer-Verlag, Berlin, Heidelberg, 174–189.