

## Article

# An Unmanned Aerial Vehicle Troubleshooting Mode Selection Method Based on SIF-SVM with Fault Phenomena Text Record

Linchao Yang <sup>1</sup>, Guozhu Jia <sup>1</sup>, Ke Zheng <sup>1</sup>, Fajie Wei <sup>1</sup>, Xing Pan <sup>2</sup>, Wenbing Chang <sup>2</sup> and Shenghan Zhou <sup>2,\*</sup> 

<sup>1</sup> School of Economics and Management, Beihang University, Beijing 100191, China; yanglinchao@buaa.edu.cn (L.Y.); jiaguozhu@buaa.edu.cn (G.J.); zhengke@buaa.edu.cn (K.Z.); weifajie@buaa.edu.cn (F.W.)

<sup>2</sup> School of Reliability and Systems Engineering, Beihang University, Beijing 100191, China; panxing@buaa.edu.cn (X.P.); changwenbing@buaa.edu.cn (W.C.)

\* Correspondence: zhoush@buaa.edu.cn

**Abstract:** At present, the research on fault analysis based on text data focuses on fault diagnosis and classification, but it rarely suggests how to use that information to troubleshoot faults reported in unmanned aerial vehicles (UAVs). Selecting the exact troubleshooting procedure to address faults reported by UAVs generally requires experienced technicians with professional equipment. To improve the efficiency of UAV troubleshooting, this paper proposed a troubleshooting mode selection method based on SIF-SVM (Serial information fusion and support vector machine) using the text feature data from fault description records. First, Word2Vec was used in text data feature extraction. Second, in order to increase the amount of information in the modeling data, we used the information fusion method. SVM was then used to construct the classification model for troubleshooting mode selection. Finally, the effectiveness of the proposed model was verified by using the fault record data of a new fixed-wing UAV.



**Citation:** Yang, L.; Jia, G.; Zheng, K.; Wei, F.; Pan, X.; Chang, W.; Zhou, S. An Unmanned Aerial Vehicle Troubleshooting Mode Selection Method Based on SIF-SVM with Fault Phenomena Text Record. *Aerospace* **2021**, *8*, 347. <https://doi.org/10.3390/aerospace8110347>

Academic Editor: Daniel Ossmann

Received: 14 September 2021

Accepted: 13 November 2021

Published: 15 November 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

With the continuous development of unmanned aerial vehicles (UAV), their function and structure are becoming more and more complex. In order to reduce the cost, UAVs usually have low redundancy designs, so the failure rate of UAVs is higher than that of manned aircraft. When a UAV has a fault or generates an early warning, the on-site maintenance personnel usually locate the faulty system according to the warning, and then decide on which maintenance method to adopt. However, in this process, a large number of tests need to be carried out, and professional personnel and equipment are needed. Especially for young and inexperienced maintenance personnel, it can be difficult to determine the maintenance method according to the fault phenomenon. Therefore, using accumulated historical data to develop a preliminary troubleshooting method for locating system faults indicated by the UAV early warning system would improve maintenance efficiency, reduce costs, and improve failure rates.

At present, the research on UAV fault based on data drive have focused on fault diagnosis, fault prediction, fault classification, etc. However, in real-world UAV maintenance scenarios, fault diagnosis requires experienced technicians and professional equipment to locate the fault and then judge how to resolve it. The UAV data records fully describe any faults reported and what actions were taken to address them. By using these historical data, this study aimed to develop a data-driven approach for preliminarily troubleshooting UAV early fault warnings according to the actual UAV maintenance process.

The proposed model would be able to locate the fault in the system according to the early warning signal and offer suggestions on how to resolve it. We used Word2Vec to extract the features of UAV fault text data. In order to improve the accuracy of the

troubleshooting, we adopted an information-fusion method to integrate the fault location information and fault phenomenon information and used the SVM model to determine which troubleshooting method to employ according to actual UAV maintenance scenarios. Moreover, we used the fault record data from a new fixed-wing UAV to verify the proposed method. The rest of this paper is organized as follows. Section 2 presents a literature review of data driven fault analysis methods and text feature extraction. Section 3 introduces the text data preprocessing method, the Word2Vec feature extraction method, and SIF-SVM. Section 4 discusses the process of selecting troubleshooting modes by using SIF-SVM based on the text data of UAV fault phenomenon description. Section 5 describes the experiment in this paper and analyzes the experimental results. In Section 6, the conclusions are given.

## 2. Literature Review

### 2.1. Data Driven Fault Analysis Methods

With increased attention paid to data and the development of machine learning technology, data-driven fault analysis methods have been developed [1–3]. In the field of aircraft, some researchers have used data-driven methods to diagnose, predict, and analyze aircraft faults. Aouaouda et al. [4] discussed a fault estimation and robust control law reconstruction technology. Shen et al. [5] developed another data-driven fault diagnosis framework to monitor the state of aviation gas turbine engines by using a hybrid multi-mode machine learning strategy. Nguyen et al. proposed a fault-tolerant control method based on adaptive, nonsingular fast terminal sliding mode control and neural network approximation for four-axis aircraft. In addition, an external disturbance sliding mode control method was proposed for the altitude attitude system [6]. Yang et al. [7] used interval data to reduce the dimension of flight data and realized real-time prediction of UAV faults through a BP neural network. Zheng et al. [8] proposed a composite fault-marking method based on UAV flight data and BIT recorded data and diagnosed the UAV composite fault mode by using XGBoost, LightGBM, and a modified CNN algorithm. Nguyen et al. [9] proposed a fault-tolerant control method based on adaptive nonsingular fast terminal sliding mode control and neural network approximation, which could control the disturbance of four-axis aircraft.

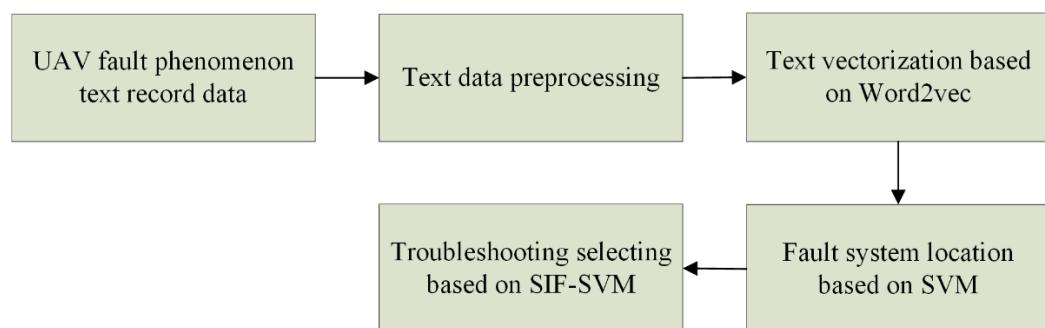
### 2.2. Text Feature Extraction

In addition to using structured numerical data, some researchers have used unstructured text data to analyze and study the faults of complex equipment. Using text data to analyze faults is usually divided into two steps. First, the text is preprocessed and feature extracted to transform the text into a numerical form that can be understood by the computer. Text feature extraction is the key step of using text data for analysis. The commonly used text feature extraction methods include TF-IDF, LDA, Word2Vec, and Doc2Vec [10–13]. For different varied scenes, researchers may use varied feature extraction methods, which may be related to the different text features in each real scene: the fault text records in some scenes are better represented by keywords, whereas some scenes rely more on semantic analysis. Researchers usually choose appropriate methods for modeling by comparing the accuracy of different text feature extraction methods and machine learning methods [11,14,15]. Second, according to the extracted text features, they build an appropriate model for fault diagnosis, prediction, or other analysis. The selection of a machine learning algorithm is the same as that for structured data, which need to be selected according to the characteristics of the research object. Wang et al. [15] used a priori LDA model to extract the features of high-speed railway fault text records and established a fault location model based on SVM. By comparing the accuracy of classification models, it was found that the potential LDA model was better than the TF-IDF and traditional LDA models used in fault text feature extraction for high-speed railways. Chang et al. [16] used a Word2Vec moving-distance model to analyze fault text, used the K-means algorithm to cluster typical faults, and then used the PrefixSpan sequence mining algorithm to mine fault sequence information, and finally trained a Bayesian fault network model to predict

aircraft faults. Li et al. [17] processed the text data of typical power grid fault cases, established a corpus and transformed it into TF-IDF frequency matrix, clustered it using the K-means algorithm based on a Calinski-Harabaz index, and established the mapping table of fault information and solutions. Xu et al. [14] introduced expert fault knowledge through a cloud-similarity measurement, improved the CNN model, and proposed a text driven aircraft fault diagnosis model based on a word-based and prior-knowledge CNN to diagnose aircraft faults through aircraft maintenance log data.

### 3. Methods

Compared with structured numerical data, UAV fault phenomenon text data belong to unstructured data. Text data are unstructured and cannot be directly used to establish a mathematical model for calculation, and the text records of different fields and business objects vary widely. To use the text data of UAV fault phenomenon to select a troubleshooting mode, the text data were preprocessed first, including text word segmentation, the removal of “stop words”, etc. Second, the preprocessed text data were vectorized to obtain the text vector data of UAV fault phenomenon. Third, SVM model was trained by using fault phenomenon text vector data, and the fault locations were determined. Fourth, SIF-SVM was used to fuse the information of the fault location and the fault phenomenon, and the troubleshooting mode was selected. The preprocessing and modeling process of text recording data of UAV fault phenomenon is shown in Figure 1.



**Figure 1.** UAV fault phenomenon text record data processing and modeling process.

#### 3.1. Text Data Preprocessing

In text representation, there are spaces between English words that separate the words. Unlike English, there is no space between Chinese words. Because text records are composed of words and words are the smallest sequence with meaning in text, determining the individual words used would be crucial in making the text data usable. To determine individual Chinese words, researchers focused on dictionary-based word-segmentation methods, machine learning methods, and deep learning methods. The selection of a Chinese word segmentation method based on machine learning directly determines the accuracy and requires computing resources, and the deep learning method requires even more resources. For Chinese word segmentation in a specific field, a word segmentation method based on the professional dictionary for that field will have the best efficiency and accuracy.

The method of word segmentation using a known dictionary was based on the existing Chinese dictionary and compared the text with the dictionary to generate acyclic graphs of the sentences, intercepting and segmenting the text sentences after finding the shortest path. However, if a word was not registered in the dictionary, it would cause an error. Therefore, in this study, based on expert experience and professional knowledge, we have established a professional dictionary in the field of UAVs (as shown in Table 1) and used that dictionary to segment the text data.

**Table 1.** Example of UAV fault domain professional dictionary based on expert experience and professional knowledge.

Words in UAV Fault Field
zero calibration switch, differential signal, backup ring, gearbox, single-phase transformer, IFF system, electric soldering iron, steering gear, protective sticker, heading gyro, angular velocity gyro, throttle, air box, coupling, ...

The most commonly used Chinese word segmentation tool is Jieba. For example, Xiao et al. [18] used Jieba to segment Weibo data and identify urban storm disasters; Chang et al. [16] used Jieba to mine UAV fault record text; Qiu et al. [19] used Jieba to segment legal text data. The Jieba word segmentation tool can add custom words, identify parts of speech, extract keywords, and identify unlisted words.

“Stop words” refers to functional words or high-frequency words, such as auxiliary words, prepositions, conjunctions, modal particles, and so on, which frequently appear in various documents and add little information or context. First, eliminating these words can effectively reduce the resource consumption of feature extraction and text mining, reduce the storage space of the system, and improve the operation efficiency. Second, it can reduce the impact of stop words on the subsequent establishment of prediction models. Therefore, these words are automatically filtered before text vectorization, feature extraction, and text mining.

### 3.2. Word2Vec

Word2Vec is a text feature extraction method that has developed rapidly in recent years. Word2Vec is based on the word distribution hypothesis, which suggests that the semantics of a word are determined by its context, and the semantics of words with similar contexts are also similar [20,21]. The word distribution hypothesis holds that whether the word can be better represented is mainly determined by the modeling of the contextual information. The most commonly used statistical model is the n-gram model. For a sentence  $(w_1, w_2, \dots, w_i, \dots, w_m)$  composed of words, the possibility of this sentence can be expressed as:

$$P(w_1 w_2 \dots w_n) = P(w_1) P(w_2|w_1) \dots P(w_i|w_1, \dots, w_{i-1}) \dots P(w_m|w_1, \dots, w_{m-1}) \quad (1)$$

Based on the maximum likelihood estimation of relative frequency, the conditional probability  $P(w_i|w_1, \dots, w_{i-1})$  of  $i$ th word can be expressed as:

$$P(w_i|w_1, \dots, w_{i-1}) = \frac{\text{count}(w_1, \dots, w_i)}{\text{count}(w_1, \dots, w_{i-1})} \quad (2)$$

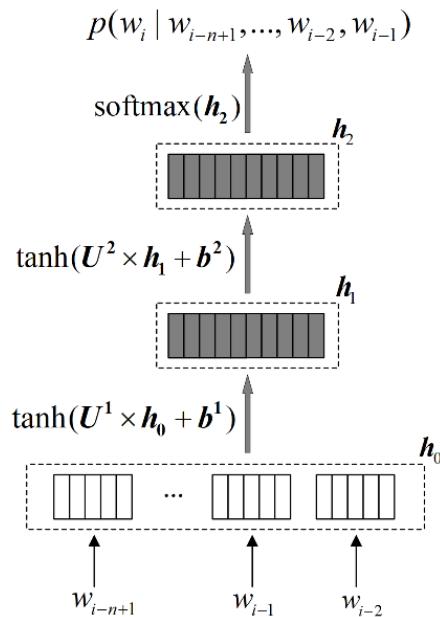
According to the Markov chain modeling, it is assumed that the current word only depends on the previous  $n - 1$  words, and its occurrence probability is

$$P(w_i|w_1, \dots, w_{i-1}) = P(w_i|w_{i-n+1}, \dots, w_{i-1}) \quad (3)$$

Therefore, we only need to model  $P(w_i|w_{i-n+1}, \dots, w_{i-1})$  to get the representation of the  $i$ th word. To solve this problem, Bengio et al. [22] proposed a language model based on a feedforward neural network. Its premise was that each word is mapped to a low dimensional continuous real vector, and the probability of  $p$  in an n-gram model is modeled in the continuous vector space. For a three-layer feedforward neural network (Figure 2), the first  $n - 1$  words are mapped into word vectors and spliced into  $h_0$ :

$$h_0 = [e(w_{i-n+1}); \dots; e(w_{i-1})] \quad (4)$$

where  $e(w_{i-1}) \in R^d$  represents the  $d$ -dimensional word vector corresponding to word  $w_{i-1}$ . The word vector matrix  $L$  can be initialized randomly at the beginning and optimized as a parameter during model training.



**Figure 2.** Feedforward neural network language model.

$h_0$  learns the expression of  $n - 1$  words through the hidden layer:

$$h_0 = f(\mathbf{U}^1 \times h_0 + \mathbf{b}^1) \quad (5)$$

$$h_1 = f(\mathbf{U}^2 \times h_1 + \mathbf{b}^2) \quad (6)$$

where the nonlinear function  $f(\bullet) = \tanh(\bullet)$  can be selected as the activation function. Finally, the probability distribution of each word in thesaurus  $V$  is calculated by softmax function:

$$P(w_i | w_{i-n+1}, \dots, w_{i-1}) = \frac{\exp\{h_2 \cdot e(w_i)\}}{\sum_{k=1}^{|V|} \exp\{h_2 \cdot e(w_k)\}} \quad (7)$$

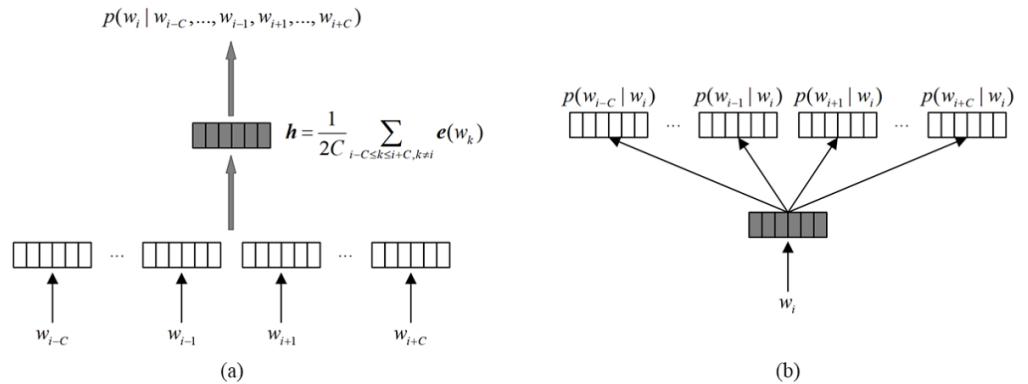
In model training,  $\mathbf{U}^1$ ,  $\mathbf{U}^2$ ,  $\mathbf{b}^1$ ,  $\mathbf{b}^2$ , and  $L$  are regarded as the parameters  $\theta$  of the feedforward neural network. The goal of model training is to optimize  $\theta$  and maximize the log likelihood of the training set:

$$\theta^* = \underset{\theta}{\operatorname{argmax}} \sum_{m=1}^M \log P(w_{i1}^{m_i}) \quad (8)$$

The Word2Vec model was based on the language model proposed by Bengio et al. The following improvements were made on the basis of the neural network language model: first, Word2Vec ignored words that appeared less frequently in the corpus; second, the input layer used word vector summation instead of word vector splicing, which reduces the dimension of the input layer. Third, the neural network model was improved, and the setting of the hidden layer was cancelled. Fourth, the hierarchical softmax function based on a Huffman coding tree was used as the output layer. Fifth, the negative-sampling algorithm was used to randomly select negative samples for updating in the training process.

Word2Vec improved the feedforward neural network language model and used the continuous bag of words (CBOW) and skip-gram models. The CBOW model predicts the

current word by inputting the context word, and the skip-gram model uses the current word to predict the context word, as shown in Figure 3. Of the two models, the skip-gram model has more stringent requirements for training data: the corpus must be sufficient, the number of words in it must be large enough, and it must include as many sentences as possible that reflect the relationship between words.



**Figure 3.** Schematic diagram of CBOW model and skip-gram model. (a) CBOW model. (b) Skip-gram model.

### 3.3. SIF-SVM

In the classification task of machine learning, the objects need to be reasonably classified according to the known information. Therefore, adding useful information is an effective way to improve the model. Therefore, integrating independent variables by means of information fusion when constructing the independent variables of the input model will improve the efficiency and accuracy of the machine learning model. The SIF-SVM constructed in this paper used serial information fusion to construct new independent variables during the independent variable stage, and then select the troubleshooting method through the SVM model.

#### 3.3.1. Serial Information Fusion

The serial information fusion method combines two separate feature vectors into a joint vector [23]. This method can effectively increase the information content of sample data [24].

Defining the sample space  $\Omega$ ,  $A$  and  $B$  are the feature spaces of different angles defined in the sample space. The corresponding two feature vectors  $\alpha \in A$ ,  $\beta \in B$  and  $\alpha$  are  $n$ -dimensional vectors and  $\beta$  are  $m$ -dimensional vectors. After serial information fusion, the following results are obtained:

$$\gamma = \begin{pmatrix} \alpha \\ \beta \end{pmatrix} \quad (9)$$

where  $\gamma$  is a  $(n + m)$ -dimensional vector.

#### 3.3.2. Support Vector Machine

Support vector machine (SVM) is one of the commonly used supervised learning algorithms in the field of machine learning, which was proposed by Cortes and Vapnik in 1995 [25]. In a classification problem, SVM finds a hyperplane in the sample space and uses the hyperplane to divide the samples according to different categories, which has proven to yield good performance and advantages. SVM has been shown to work well on high-dimensional and small sample data and is widely used in the field of fault diagnosis [26–28].

According to the basic principle of a support vector machine, for a given training sample set  $D = \{(x_1, y_1), (x_2, y_2), \dots, (x_m, y_m)\}$ , where  $y_i \in \{-1, +1\}$ , we need to find a

hyperplane in the sample space to divide the training sample set  $D$ . In the sample space, the partition hyperplane can be expressed as:

$$\mathbf{w}^T \mathbf{x} + b = 0 \quad (10)$$

where  $\mathbf{w} = (w_1; w_2; \dots; w_d)$  is the normal vector, which determines the direction of the hyperplane used to divide the sample; and  $b$  is the displacement term, which determines the distance between the hyperplane and the origin.

If the hyperplane can divide the sample correctly, for  $(\mathbf{x}_i, y_i) \in D$ :

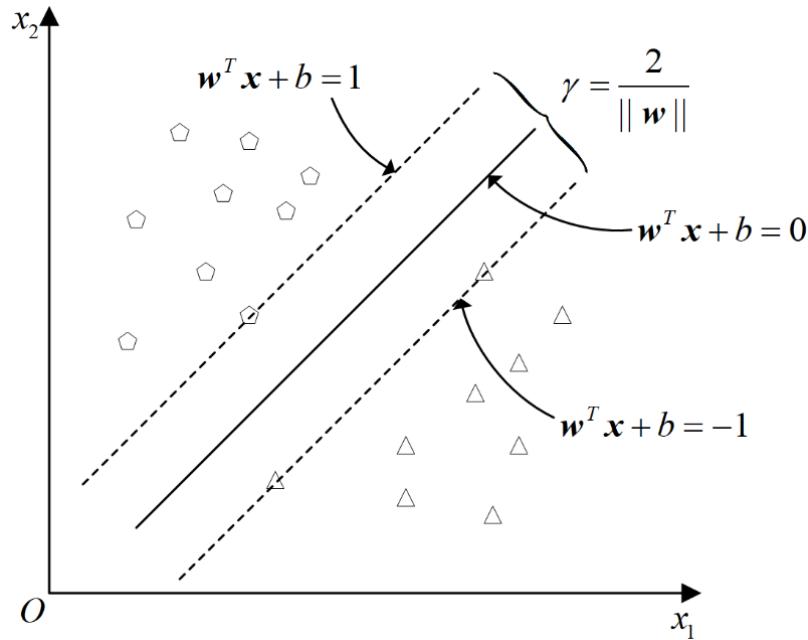
$$\begin{cases} \mathbf{w}^T \mathbf{x}_i + b > 0, y_i = +1 \\ \mathbf{w}^T \mathbf{x}_i + b < 0, y_i = -1 \end{cases} \quad (11)$$

Based on (11), there is always scaling transformation, so that the following formula is true:

$$\begin{cases} \mathbf{w}^T \mathbf{x}_i + b \geq +1 \\ \mathbf{w}^T \mathbf{x}_i + b \leq -1 \end{cases} \quad (12)$$

As can be seen from Figure 4, the equal sign of the above formula was established according to the several sample points closest to the hyperplane that did not belong to the same class. These sample points are called support vectors, and the sum of the distances between the two support vectors that do not belong to the same class and the hyperplane is called the interval, which can be expressed as:

$$\gamma = \frac{2}{\|\mathbf{w}\|} \quad (13)$$



**Figure 4.** Schematic diagram of support vector machine.

When the interval reaches the maximum, the sample can be divided optimally. Therefore, the main task was to find the partition hyperplane with the largest interval from the support vector; that is, to find the appropriate constraint parameters  $\mathbf{w}$  and  $b$  to maximize  $\gamma$ . The problem can be described as:

$$\begin{aligned} & \max_{\mathbf{w}, b} \frac{2}{\|\mathbf{w}\|} \\ & \text{s.t. } y_i(\mathbf{w}^T \mathbf{x}_i + b) \geq 1, i = 1, 2, \dots, m \end{aligned} \quad (14)$$

Equation (14) can be converted to:

$$\begin{aligned} & \max_{\mathbf{w}, b} \frac{1}{2} \|\mathbf{w}\|^2 \\ & \text{s.t. } y_i(\mathbf{w}^T \mathbf{x}_i + b) \geq 1, i = 1, 2, \dots, m \end{aligned} \quad (15)$$

By solving the convex quadratic programming problem, the model corresponding to the maximum interval partition hyperplane can be obtained:

$$f(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + b \quad (16)$$

However, in practical applications, we often have high-dimensional complex samples that cannot be divided directly by linearity. For the linearly nonseparable problem, the idea of SVM is to map the samples to a higher dimensional feature space, so that the samples can be linearly separable in this space. For the finite dimensional sample space, there must be a high-dimensional feature space that can realize the division of samples.

The mapped eigenvector of  $\mathbf{x}$  is expressed as  $\phi(\mathbf{x})$ , (16) and can be expressed as:

$$f(\mathbf{x}) = \mathbf{w}^T \phi(\mathbf{x}) + b \quad (17)$$

where  $\mathbf{w}$  and  $b$  are model parameters. For (17),  $\phi(\mathbf{x})$  can be used to rewrite the equation as:

$$\begin{aligned} & \max_{\mathbf{w}, b} \frac{1}{2} \|\mathbf{w}\|^2 \\ & \text{s.t. } y_i(\mathbf{w}^T \phi(\mathbf{x}_i) + b) \geq 1, i = 1, 2, \dots, m \end{aligned} \quad (18)$$

The dual problem of (18) is

$$\begin{aligned} & \max_{\alpha} \sum_{i=1}^m \alpha_i - \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^m \alpha_i \alpha_j y_i y_j \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j) \\ & \text{s.t. } \sum_{i=1}^m \alpha_i y_i = 0, \\ & \alpha_i \geq 0, i = 1, 2, \dots, m \end{aligned} \quad (19)$$

Let  $\kappa(\mathbf{x}_i, \mathbf{x}_j) = \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle = \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j)$ , then (19) can be written as:

$$\begin{aligned} & \max_{\alpha} \sum_{i=1}^m \alpha_i - \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^m \alpha_i \alpha_j y_i y_j \kappa(\mathbf{x}_i, \mathbf{x}_j) \\ & \text{s.t. } \sum_{i=1}^m \alpha_i y_i = 0, \\ & \alpha_i \geq 0, i = 1, 2, \dots, m \end{aligned} \quad (20)$$

By solving (20), we obtain:

$$\begin{aligned} f(\mathbf{x}) &= \mathbf{w}^T \phi(\mathbf{x}) + b \\ &= \sum_{i=1}^m \alpha_i y_i \phi(\mathbf{x}_i)^T \phi(\mathbf{x}) + b \\ &= \sum_{i=1}^m \alpha_i y_i \kappa(\mathbf{x}_i, \mathbf{x}) + b \end{aligned} \quad (21)$$

where  $\kappa(\cdot, \cdot)$  is the kernel function of the support vector machine model, and the inner product of  $\mathbf{x}_i$  and  $\mathbf{x}_j$  in the feature space is equal to the result of  $\mathbf{x}_i$  and  $\mathbf{x}_j$  calculated by kernel function  $\kappa(\cdot, \cdot)$  in the original sample space. However, the specific form of the feature vector  $\phi(\cdot)$  that maps the sample to the high-dimensional space cannot be determined

directly, so the specific form of kernel function  $\kappa(\cdot, \cdot)$  cannot be obtained, so we can only select the commonly used kernel function for representation. At present, a radial basis kernel function is the most commonly used:

$$\kappa(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right) \quad (22)$$

where  $\sigma$  is the bandwidth of the radial basis function and  $\sigma > 0$ .

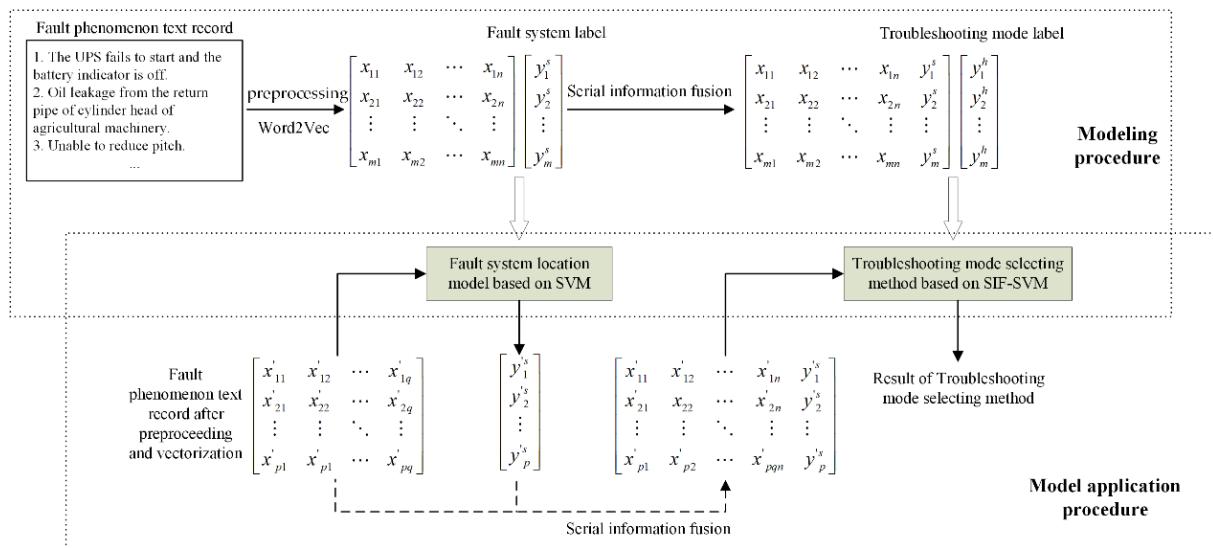
#### 4. Application

In case of UAV failure or abnormality, the field maintenance teams usually need to locate the fault according to the phenomenon and determine the troubleshooting mode based on the location and damage. During fault analysis, a series of scene tests need to be carried out that must be implemented by professional technicians with testing equipment. Based on the fault phenomenon, the method for selecting a troubleshooting mode was established. For the faults that could be handled and eliminated on site, the fault detection process would be omitted to avoid delays in the flight mission.

In real-world UAV maintenance, the troubleshooting mode is determined after fault location. Therefore, in the modeling, the serial information fusion method was used to fuse the fault phenomenon and location. This not only conformed to the logic of the practical work, but also increased the useful information, which, in theory, should improve the effect of the model.

In the modeling process, the fault phenomenon record text data were preprocessed first, and Word2Vec was used for vectorization. For  $m$  fault phenomenon text records, an independent variable matrix  $X_{m \times n}$ , where  $n$  is the length of vectorized text vector. Second, based on the SVM model, the fault location model was established by using  $X_{m \times n}$  and fault location label vector  $y_m^s = [y_1^s, y_2^s, \dots, y_m^s]^T$ . Third,  $X_{m \times n}$  and  $y_m^s$  are fused to obtain the new independent variable matrix  $[X_{m \times n}, y_m^s]$  of fused information. Fourth, using  $[X_{m \times n}, y_m^s]$  and troubleshooting mode label vector  $y_m^h = [y_1^h, y_2^h, \dots, y_m^h]^T$ , the troubleshooting mode selection model was established based on the SVM model.

Based on the UAV fault record text data, the UAV fault location model and troubleshooting mode selection model were established. In the modeling, the text description of fault phenomenon was preprocessed, and feature extraction was carried out. Using the text feature data and fault location label, the fault location model was then established via the SVM. Next, the serial information fusion method was used to fuse the fault phenomenon text feature data with the fault location labels, and, thus, the troubleshooting mode selection model was established by using the troubleshooting mode labels and the SVM. In the application of the model, through the fault location model, the fault location was determined by using the text feature data from the fault phenomenon. Once the fault location and fault phenomenon text were fused, the fused data were used to select the troubleshooting mode through the troubleshooting mode selection model. The modeling procedure and model application procedure are shown in Figure 5.



**Figure 5.** Modeling and application flow of UAV troubleshooting mode selection method based on SIF-SVM.

## 5. Experiments and Results

In order to verify the effectiveness of the troubleshooting mode selection method based on SIF-SVM that has been proposed in this paper, we used the fault record text data accumulated by a new fixed-wing UAV. The data were recorded in Chinese with a total of 662 samples. Some data are shown in Table 2. The recorded faults involved four systems: the ground system (1), the power system (2), the flight control system (3), and the aircraft platform (4). According to the UAV troubleshooting mode selection modeling based on SIF-SVM, the experiment mainly included fault phenomenon text data preprocessing, text feature extraction based on Word2Vec and model construction based on SIF-SVM.

**Table 2.** Examples of fault phenomenon recording text data.

Text Number	Fault Phenomenon Text Record	Fault System
1	UPS failed to start; ALARM lamp given an alarm.	ground system (1)
2	During the engine test run, the engine works abnormally with surge.	power system (2)
3	Vertical gyro-pitch out of tolerance, pitch angle instability.	flight control system (3)
4	The brake valve was hot and had a paste smell, and the preset brake volume has no response.	aircraft platform (4)
...	...	...

The original data are recorded in Chinese, so the data shown in this table were translated into English.

### 5.1. Preprocessing of Fault Phenomenon Text Record

Because the collected fault phenomenon text data of the new fixed-wing UAV were recorded in Chinese, word segmentation and stop word removal were completed before analysis. We used Jieba for word segmentation, and used our professional UAV dictionary. The stop words were then removed from the word segmentation results. The above work was done using Jieba, version 0.42.1, in Python, version 3.8.5. The fault phenomenon text record data after preprocessing are shown in Table 3.

**Table 3.** Examples of fault phenomenon recording text data after preprocessing.

Text Number	Fault Phenomenon Text Record	Fault System
1	UPS/failed/start/ALARM lamp/gived/alarm	ground system (1)
2	engine/test/run/engine/works/abnormally/surge	power system (2)
3	vertical/gyro/pitch/out/tolerance/pitch/angle/instability	flight control system (3)
4	brake/valve/hot/had/paste/smell/preseted/brake volume/has/no/response	aircraft platform (4)
...	...	...

The original data were recorded in Chinese, so the data shown in this table were translated into English.

### 5.2. Text Feature Extraction Based on Word2Vec

Because a computer cannot directly use the text data for calculation, it was necessary to convert the preprocessed fault phenomenon descriptions from the text data records into structured numerical data in order to build a machine learning model. For the feature extraction of the text data, we used the Word2Vec model. Word2Vec can obtain the word vector of each word in the corpus. We summed the word vectors of all words in the text to get the text vector. In addition, Word2Vec can select the dimension of a word vector, which is conducive to the dimensionality reduction of text features, and can investigate the feature expressions under different dimensions for different text databases. We used Genism, version 4.0.1, in Python, version 3.8.5, to extract text features based on Word2Vec; a selection of the results is shown in Table 4.

**Table 4.** UAV fault phenomenon recording text feature vector extracted by Word2Vec.

Number	Dimension									
	1	2	3	4	5	...	98	99	100	
1	−0.011	0.010	0.039	0.030	0.043	...	0.046	−0.016	−0.023	
2	−0.016	−0.018	0.010	0.010	0.008	...	0.015	0.008	−0.025	
3	−0.023	−0.006	−0.001	0.041	0.027	...	0.007	−0.015	−0.020	
4	−0.009	0.009	−0.007	0.008	0.001	...	−0.012	0.002	−0.006	
...	...	...	...	...	...	...	...	...	...	...

### 5.3. UAV Troubleshooting Mode Selection Based on SIF-SVM

According to the modeling process proposed above, in order to select the correct troubleshooting mode to respond to fault signaling by the UAV, we had to first establish the fault location model. Using the information fusion method, the fault location and fault phenomenon text features were then fused, and the troubleshooting mode selection model was established. The independent variable data set based on information fusion is shown in Table 5.

**Table 5.** Independent variable data set integrating fault phenomenon and fault location information.

Number	$y^s$	$x_1$	$x_2$	$x_3$	$x_4$	...	$x_{98}$	$x_{99}$	$x_{100}$
1	1	−0.011	0.010	0.039	0.030	...	0.046	−0.016	−0.023
2	2	−0.016	−0.018	0.010	0.010	...	0.015	0.008	−0.025
3	3	−0.023	−0.006	−0.001	0.041	...	0.007	−0.015	−0.020
4	4	−0.009	0.009	−0.007	0.008	...	−0.012	0.002	−0.006
...	...	...	...	...	...	...	...	...	...

According to the complexity of actual UAV troubleshooting modes and expert experience, the independent variables were divided into three categories and labeled, as shown in Table 6. As a result, the troubleshooting selection mode could be simplified, and preliminary troubleshooting procedures could be provided to field maintenance personnel.

**Table 6.** Classification of UAV troubleshooting modes and label setting.

Troubleshooting Mode	Label
Large parts need to be replaced or the UAV needs to return to the factory for maintenance	1
Small accessories on the UAV need to be replaced, which can be completed on the mission site	2
The UAV faults can be eliminated through basic methods, such as on-site vehicle service, grinding, gluing, cleaning, and assembly	3

In order to verify the effectiveness of the troubleshooting mode selection method based on the SIF-SVM, both in theory and in practice, we constructed three models for comparison, as shown in Table 7. By comparing the effects of Model 1 and Model 2, we theoretically verified the effectiveness of the SIF-SVM; using Model 3, we verified the effectiveness of the SIF-SVM under real-world conditions.

**Table 7.** Description of models setting.

Model	Description
Model 1	The troubleshooting modes was trained and tested by using only the fault phenomenon text feature vector.
Model 2	The real fault location information was fused with the fault phenomenon text feature vector, and the troubleshooting modes were trained and tested.
Model 3	Built the model according to the SIF-SVM process proposed in this paper. First, only the fault phenomenon text feature vector in the training set was used to train the fault location model, and then the training set was fused with the real fault location information to train the troubleshooting mode selection model. According to the real world process, in the model test, we first used the fault phenomenon text data to locate the fault. After obtaining the location results, we integrate the location results into the test set and then selected the troubleshooting mode.

We used the *e1071* package (version 1.7) in R (version 1.0.3) to build the SVM model on a computer with the AMD Ryzen 7 1700 Eight-Core Processor, 3 GHz CPU, and 32 GB RAM. We randomly selected two-thirds of the data to train the model and used the remaining data as the test set to verify the predictive ability of the model. We selected accuracy, precision, recall, and  $F_1$ -score as the evaluation indicators of the model effect. Because this paper aimed to solve a multi-classification problem, we use the calculation method of the multi-classification problem to calculate these indexes [29]. In order to compare the Word2Vec text feature extraction of UAV fault records, we also used the popular TF-IDF, LDA, and Doc2Vec models for text feature extraction and established the model. Moreover, we also compare the effects of different text vector dimensions on the model.

As can be seen from Figure 6, the troubleshooting mode selection model (Model 2 and Model 3) constructed based on the SIF-SVM was better at determining the fault troubleshooting mode through the fault phenomenon description. The comparison of the two models showed that the location of the fault was necessary information for the UAV troubleshooting mode selection. After adding the location information of the fault, the model's selection was improved, and troubleshooting mode selection based on the reported fault was more accurate. In the comparison of Model 1 with Model 3, we found that in a real-world UAV maintenance scenario, using the fault phenomenon to locate the fault, and then using the positioning results for information fusion, after which the troubleshooting mode was selected, was a more effective method than determining the troubleshooting mode only through the fault phenomenon. Because Model 3 used the

fault location information judged by the fault location model, there was a risk of error as compared to the real fault location information used by Model 2. Therefore, it had been expected that the performance of Model 3 would be slightly lower than that of Model 2.

The text features extracted by Word2Vec were effective for modeling, as can be seen from Figure 6a. In different text vector dimensions, accuracy, precision, and recall of Model 3 achieved more than 80%. When the text vector dimension was 300, the modeling effect was the best: the accuracy was 0.8733, the precision was 0.8831, and the recall was 0.8633. As shown in Figure 6b, the text features extracted by Doc2Vec were poor for modeling. The essence of LDA was subject extraction of the text, so the text vector length was set to 50, 100, 150, 200, and 250. As can be seen from Figure 6c, when the text vector length was 200, the effect of Model 3 was the best, and the accuracy, precision, and recall were 0.7738, 0.8198, and 0.7480, respectively. TF-IDF extracted text features through the importance of words in the corpus, so the length of the text vector was generally the number of words contained in the corpus. The model based on the text features of TF-IDF was effective. The accuracy, precision, and recall of Model 3 were 0.8100, 0.8971, and 0.7798, respectively.

The best modeling based on each text feature extraction method is shown in Table 8. The modeling based on Word2Vec text feature extraction was the best. The modeling of the text feature extraction based on TF-IDF was also good, and its accuracy rate reached 0.8971, but the length of text vector was much larger, as compared to that of Word2Vec. In practice, this would lead to a serious computational burden. Therefore, using Word2Vec text feature extraction, when the length of text vector is 300, is the best, most efficient option.

In the actual UAV maintenance scenario (i.e., the scenario used in Model 3), the proposed troubleshooting mode selection process was a two-stage model. Within this framework, we acquired the fault location, and then selected the troubleshooting mode. As shown in Table 9, in the fault location stage, the model accuracy, precision, and recall reached 0.9005, 0.8994, and 0.8984, respectively. It showed that the fault location model could accurately locate the fault location of the UAV. In the troubleshooting mode selection stage, the model could also accurately determine the fault handling mode. The confusion matrix of the model results of the two stages is shown in Figure 7.

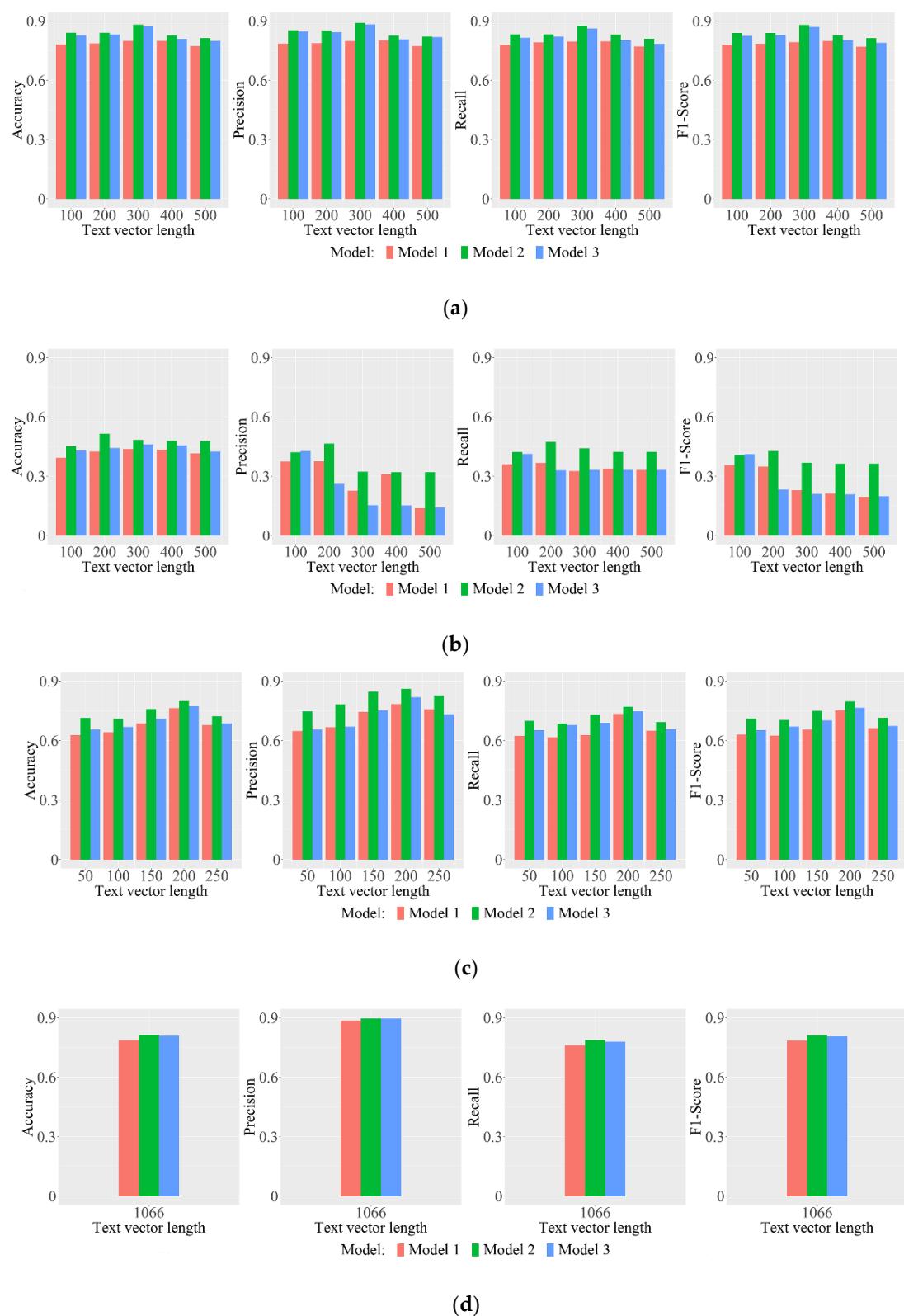
**Table 8.** Comparison of effect of troubleshooting mode selection method based on different text feature extraction methods.

Text Feature Extraction Method	Text Vector Length	Evaluation Indicators of Model 3			
		Accuracy	Precision	Recall	F <sub>1</sub> -Score
Word2Vec	300	0.8733	0.8831	0.8633	0.8711
Doc2Vec	200	0.4434	0.2612	0.3305	0.2324
LDA	200	0.7738	0.8198	0.7480	0.7658
TF-IDF	1066	0.8100	0.8971	0.7798	0.8062

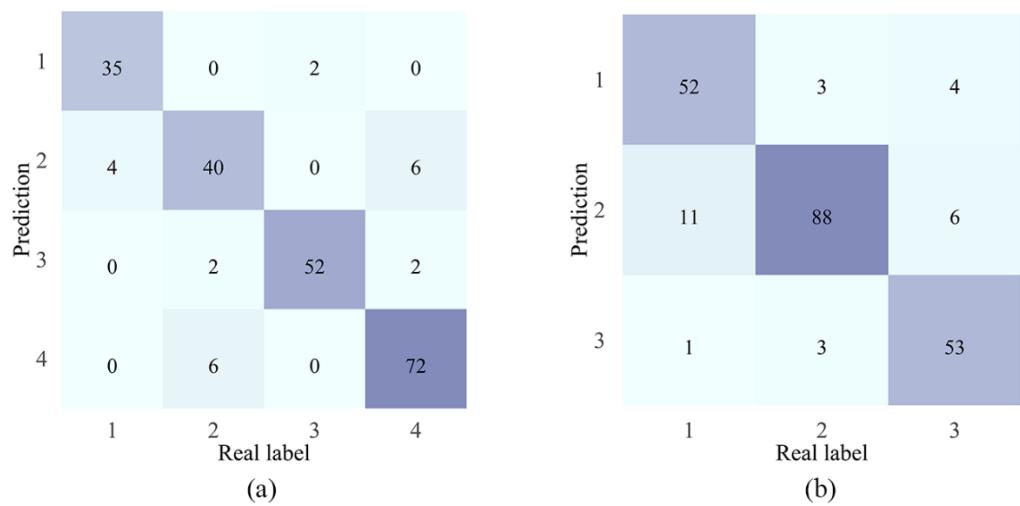
**Table 9.** The two-stage model effect of troubleshooting mode selection framework.

Stage	Accuracy	Precision	Recall	F <sub>1</sub> -Score
fault system location	0.9005	0.8994	0.8984	0.8986
troubleshooting mode selection	0.8733	0.8831	0.8633	0.8711

The text feature extraction method is Word2Vec, and the length of the text vector is 300.



**Figure 6.** Effect of troubleshooting mode selection method based on SIF-SVM (comparison of Model 1, Model 2, and Model 3). **(a)** Effect of troubleshooting mode selection method based on SIF-SVM (fault phenomenon text feature extraction based on Word2Vec). **(b)** Effect of troubleshooting mode selection method based on SIF-SVM (fault phenomenon text feature extraction based on Doc2Vec). **(c)** Effect of troubleshooting mode selection method based on SIF-SVM (fault phenomenon text feature extraction based on LDA). **(d)** Effect of troubleshooting mode selection method based on SIF-SVM (fault phenomenon text feature extraction based on TF-IDF).



**Figure 7.** Two-stage model results of confusion matrix. (The text feature extraction method is Word2Vec, and the length of the text vector is 300.) (a) Stage of fault location. (b) Stage of troubleshooting mode selection.

## 6. Conclusions

In this paper, we proposed a text-driven UAV troubleshooting mode selection method by using Word2Vec and SIF-SVM to extract the features of fault phenomenon text data and to increase the amount of information used in the modeling data, upon which we would then build a classifier. According to the results of our experiment on the fault recording text data of a new fixed-wing UAV, it was found that the proposed model could select the troubleshooting mode for the UAV fault accurately and conform to the requirements based on actual UAV maintenance processes while exceeding the capabilities of the other models without information fusion. In addition, during the fault location stage, the proposed model correctly found the actual fault location in the UAV with an outstanding performance. In future work, the method proposed in this paper could be applied to the maintenance of certain types of UAVs. Moreover, according to a large number of actual cases, the troubleshooting modes could be disseminated further to offer more detailed guidance and advanced maintenance procedures to on-site personnel.

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

**Funding:** This research is supported by the National Natural Science Foundation of China (Grant No. 71971013 & 71871003) and the Technical Research Foundation (Grant No. JSZL2016601A004). The study is also sponsored by the Fundamental Research Funds for the Central Universities (Grant No. YWF-20-BJ-J-943) and the Graduate Student Education & Development Foundation of Beihang University.

**Data Availability Statement:** Restrictions apply to the availability of these data. The data in this research came from BHUAS Technology Co., Ltd. (Hong Kong, China) Please contact Linchao Yang ([yanglinchao@buaa.edu.cn](mailto:yanglinchao@buaa.edu.cn)) to inquire about the data availability.

**Acknowledgments:** All authors would like to thank the data support by BHUAS Technology Co., Ltd.

**Conflicts of Interest:** The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

## References

1. Castellanos, M.B.; Serpa, A.L.; Biazussi, J.L.; Verde, W.M.; Sassim, N. Fault identification using a chain of decision trees in an electrical submersible pump operating in a liquid-gas flow. *J. Pet. Sci. Eng.* **2020**, *184*. [[CrossRef](#)]
2. Deng, W.; Yao, R.; Zhao, H.M.; Yang, X.H.; Li, G.Y. A novel intelligent diagnosis method using optimal LS-SVM with improved PSO algorithm. *Soft Comput.* **2019**, *23*, 2445–2462. [[CrossRef](#)]
3. Zhang, Z.; Han, H.; Cui, X.Y.; Fan, Y.Q. Novel application of multi-model ensemble learning for fault diagnosis in refrigeration systems. *Appl. Therm. Eng.* **2020**, *164*. [[CrossRef](#)]
4. Aouaouda, S.; Chadli, M. Robust fault tolerant controller design for Takagi-Sugeno systems under input saturation. *Int. J. Syst. Sci.* **2019**, *50*, 1163–1178. [[CrossRef](#)]
5. Shen, Y.Y.; Khorasani, K. Hybrid multi-mode machine learning-based fault diagnosis strategies with application to aircraft gas turbine engines. *Neural Netw.* **2020**, *130*, 126–142. [[CrossRef](#)]
6. Nguyen, N.P.; Mung, N.X.; Thanh, H.L.N.N.; Huynh, T.T.; Lam, N.T.; Hong, S.K. adaptive sliding mode control for attitude and altitude system of a quadcopter UAV via neural network. *IEEE Access* **2021**, *9*, 40076–40085. [[CrossRef](#)]
7. Yang, L.C.; Jia, G.Z.; Wei, F.J.; Chang, W.B.; Li, C.; Zhou, S.H. The CIPCA-BPNN failure prediction method based on interval data compression and dimension reduction. *Appl. Sci.* **2021**, *11*, 3448. [[CrossRef](#)]
8. Zheng, K.; Jia, G.Z.; Yang, L.C.; Wang, J.Q. A Compound fault labeling and diagnosis method based on flight data and BIT record of UAV. *Appl. Sci.* **2021**, *11*, 5410. [[CrossRef](#)]
9. Nguyen, N.P.; Mung, N.X.; Ha, L.; Huynh, T.T.; Hong, S.K. Finite-time attitude fault tolerant control of quadcopter system via neural networks. *Mathematics* **2020**, *8*, 1541. [[CrossRef](#)]
10. Abualigah, L.M.; Khader, A.T.; Hanandeh, E.S. A new feature selection method to improve the document clustering using particle swarm optimization algorithm. *J. Comput. Sci.* **2018**, *25*, 456–466. [[CrossRef](#)]
11. Curiskis, S.A.; Drake, B.; Osborn, T.R.; Kennedy, P.J. An evaluation of document clustering and topic modelling in two online social networks: Twitter and Reddit. *Inf. Process. Manag.* **2020**, *57*. [[CrossRef](#)]
12. Yamashiro, H.; Nonaka, H.; Carreon, E.C.A. Development of an unsupervised learning methods for classification of accident reports without code information. In Proceedings of the IEEE Symposium Series on Computational Intelligence (SSCI), Xiamen, China, 6–9 December 2019.
13. Zhao, Y.; Xu, T.-H.; Wang, H.-F. Text mining based fault diagnosis of vehicle on-board equipment for high speed railway. In Proceedings of the IEEE 17th International Conference on Intelligent Transportation Systems (ITSC), Qingdao, China, 8–11 October 2014.
14. Xu, Z.; Chen, B.; Zhou, S.; Chang, W.; Ji, X.; Wei, C.; Hou, W. A text-driven aircraft fault diagnosis model based on a word2vec and priori-knowledge convolutional neural network. *Aerospace* **2021**, *8*, 112. [[CrossRef](#)]
15. Wang, F.; Xu, T.-H.; Zhao, Y.; Huang, Y.-R. Prior LDA and SVM based fault diagnosis of vehicle on-board equipment for high speed railway. In Proceedings of the 18th IEEE International Conference on Intelligent Transportation Systems, Gran Canaria, Spain, 15–18 September 2015.
16. Chang, W.; Xu, Z.; You, M.; Zhou, S.; Xiao, Y.; Cheng, Y. A Bayesian failure prediction network based on text sequence mining and clustering. *Entropy* **2018**, *20*, 923. [[CrossRef](#)] [[PubMed](#)]
17. Li, G.; Zhang, Q.; Zheng, R.; Wang, C. A fault analysis method based on text clustering. In Proceedings of the IEEE 5th International Conference on Computer and Communication Systems (ICCCS), Shanghai, China, 15–18 May 2020.
18. Xiao, Y.; Li, B.; Gong, Z. Real-time identification of urban rainstorm waterlogging disasters based on Weibo big data. *Nat. Hazards* **2018**, *94*, 833–842. [[CrossRef](#)]
19. Qiu, M.; Zhang, Y.; Ma, T.; Wu, Q.; Jin, F. Convolutional-neural-network-based Multilabel Text Classification for Automatic Discrimination of Legal Documents. *Sens. Mater.* **2020**, *32*, 2659–2672. [[CrossRef](#)]
20. Firth, J. A synopsis of linguistic theory, 1930–1955. *Stud. Linguist. Anal.* **1957**, Reprint in 1962 Oxford: Basil Blackwell. 1–31.
21. Harris, Z. Distributional Structure. *Word* **1954**, *10*, 146–162. [[CrossRef](#)]
22. Bengio, Y.; Ducharme, R.; Vincent, P.; Jauvin, C. A neural probabilistic language model. *J. Mach. Learn. Res.* **2003**, *3*, 1137–1155. [[CrossRef](#)]
23. Yang, J.; Yang, J.Y.; Zhang, D.; Lu, J.F. Feature fusion: Parallel strategy vs. serial strategy. *Pattern Recognit.* **2003**, *36*, 1369–1381. [[CrossRef](#)]
24. Liu, C.J.; Wechsler, H. A shape- and texture-based enhanced fisher classifier for face recognition. *IEEE Trans. Image Process.* **2001**, *10*, 598–608. [[CrossRef](#)]
25. Cortes, C.; Vapnik, V. Support-vector networks. *Mach. Learn.* **1995**, *20*, 297. [[CrossRef](#)]
26. Chen, P.; Yuan, L.; He, Y.; Luo, S. An improved SVM classifier based on double chains quantum genetic algorithm and its application in analogue circuit diagnosis. *Neurocomputing* **2016**, *211*, 202–211. [[CrossRef](#)]
27. Duan, L.; Xie, M.; Bai, T.; Wang, J. A new support vector data description method for machinery fault diagnosis with unbalanced datasets. *Expert Syst. Appl.* **2016**, *64*, 239–246. [[CrossRef](#)]
28. Mulumba, T.; Afshari, A.; Yan, R.; Shen, W.; Norford, L.K. Robust model-based fault diagnosis for air handling units. *Energy Build.* **2015**, *86*, 698–707. [[CrossRef](#)]
29. Mesleh, A.M.d. *Support Vector Machines Based Arabic Language Text Classification System: Feature Selection Comparative Study*; Springer: Dordrecht, The Netherlands, 2008; pp. 11–16.