



Article Fault Diagnosis of Wind Turbine with Alarms Based on Word Embedding and Siamese Convolutional Neural Network

Lu Wei¹, Jiaqi Qu², Liliang Wang², Feng Liu¹, Zheng Qian^{2,*} and Hamidreza Zareipour³

- ¹ School of Electronics and Information Engineering, Beihang University, Beijing 100191, China; weilu@buaa.edu.cn (L.W.); liuf@buaa.edu.cn (F.L.)
- ² School of Instrumentation and Optoelectronic Engineering, Beihang University, Beijing 100191, China; qujiaqi@buaa.edu.cn (J.Q.); wangliliang@buaa.edu.cn (L.W.)
- ³ Department of Electrical and Computer Engineering, University of Calgary, Calgary, AB T2N1N4, Canada; hzareipo@ucalgary.ca
- * Correspondence: qianzheng@buaa.edu.cn

Featured Application: When applied to online condition monitoring, the proposed method can assist wind turbine operators in quickly identifying the types of faults that trigger alarms. Therefore, it can reduce operation and maintenance costs and downtime losses.

Abstract: Alarms generated by a wind turbine alarm system indicate the need for emergency action by operators to protect the turbine from running into risky conditions. However, it can be challenging for operators to identify the fault types that trigger alarms, particularly with few labeled fault samples. This paper proposes a novel fault diagnosis method for wind turbines with alarms that collaboratively uses labeled and unlabeled alarms to improve diagnosis accuracy. First, the proposed method distinguishes different alarm sequences using a designed Siamese convolutional neural network with an embedding layer (S-ECNN) model. Then, the fault category of an unknown alarm sequence is diagnosed based on similarity scores. Specifically, the Skip-gram model is used to mine potential relationships among alarms in unlabeled alarm sequences, and pretrained alarm vectors are obtained. In the S-ECNN model, the pretrained alarm vectors are further optimized and trained using labeled alarm sequences. The similarity scores are calculated based on the distance between the extracted discriminative features of alarm sequences. The effectiveness of the proposed method is validated using actual alarm data from a wind farm.

Keywords: wind turbines; alarms; fault diagnosis; Siamese convolutional neural network; word embedding

1. Introduction

The installed capacity of wind power in the global market in 2021 was 93.6 GW, bringing the global total capacity to 837 GW [1]. As wind turbine technology continues to evolve, sophisticated multi-MW wind turbines have been applied for onshore and offshore wind farms [2]. However, larger wind turbines have proven to develop more failures than small ones [3]. Moreover, wind farms are generally located in remote areas with a harsh operational environment, the limited accessibility of which leads to high costs for operation and maintenance (O&M). Statistics show that the O&M costs account for 10–15% of total onshore wind farm project costs [4]. For an offshore wind farm, the O&M costs account for up to 14–30% [5]. Therefore, it is vital to reduce O&M costs for enhancing the competitiveness of wind farms.

Condition monitoring and fault diagnosis of wind turbines aiming at detecting incipient faults can improve the reliability of wind turbines and reduce O&M costs [6]. Recently, many techniques have been presented and achieved some success. Vibration analysis [7–9], oil analysis [10], and strain measurement [11] have been widely studied, and are mainly



Citation: Wei, L.; Qu, J.; Wang, L.; Liu, F.; Qian, Z.; Zareipour, H. Fault Diagnosis of Wind Turbine with Alarms Based on Word Embedding and Siamese Convolutional Neural Network. *Appl. Sci.* **2023**, *13*, 7580. https://doi.org/10.3390/ app13137580

Academic Editors: Galih Bangga and Martin Otto Laver Hansen

Received: 13 May 2023 Revised: 21 June 2023 Accepted: 26 June 2023 Published: 27 June 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https:// creativecommons.org/licenses/by/ 4.0/). used to monitor the highest-cost subcomponents of wind turbines, (e.g., main bearing, gearbox, and electric generators) due to the costs associated with mounting additional sensors and maintaining. On the other hand, supervisory control and data acquisition (SCADA) systems have become a standard installation on large wind turbines, which provide a wide range of wind turbines' operational signals. As a potentially low-cost and wide-coverage solution, plentiful studies using SCADA data for fault diagnosis have been developed [12–14]. In addition, analysis of alarms generated by wind turbine alarm systems is a promising way of fault diagnosis. Typically, alarms are triggered and recorded when key component signals exceed threshold limits [15], which indicates the need for the operator's emergency action to protect a wind turbine from running into risky conditions. Alarm systems are critically important for the safety and efficiency of wind turbines. Due to the high requirements for condition monitoring of modern large wind turbines, more and more alarm configurations are added to alarm systems, which can provide a large number of alarm data that cover almost all wind turbine subcomponents. The performance of a wind turbine can be monitored through a proper analysis of these collected alarms.

However, it is not easy for on-site operators to diagnose wind turbine faults through alarms. Alarms typically contain descriptive information about an abnormal situation, which cannot directly indicate the fault types. Moreover, large numbers of alarms are usually triggered in a short period once a specific fault occurs. The operator is easily overwhelmed by these alarms because it exceeds his response capability. There are three main reasons for the situation. First, irrational and redundant alarm configurations commonly exist [16] in alarm systems, which will cause false alarms and repeated alarms. Second, modern turbines present a high level of interconnectivity due to the mechanical structures, electrical connections, and complex control systems [17]. The propagation of faults in wind turbines will trigger many consequential alarms and related alarms [18]. Third, the operating conditions of wind turbines are complex and changeable. Under different operating conditions, the same fault could trigger different alarms [19]. As a result, when overwhelmed by alarms, the operator needs to rely on extra expert consultation for fault analysis.

Some researchers have focused on the use of alarms for wind turbine fault diagnosis. A feasibility study of the wind turbine alarm diagnosis method using an artificial neural network was presented in [20]. To find alarm patterns, the alarms triggered by a fault were transformed into an alarm matrix. However, the actual fault samples are difficult to satisfy its exponential dependence on data volume. A time-sequence method and probabilitybased method were proposed in [15] for analyzing alarms. The fault cases on the wind turbine converter and pitch system were used to verify the proposed methods. The results showed that both methods had the potential to rationalize alarm data and identify fault locations. However, the issue of time consumption must be solved when the methods are applied to larger data. An improved Apriori algorithm was proposed in [21] to analyze alarms, which occurred during blade angle asymmetry fault. The results showed that the related alarms could be integrated into one critical alarm to reduce the number of alarms. The accuracy of the method is limited due to its dependence on sufficient sample data. A clustering analysis of alarm sequences for characterizing and classifying wind stoppages was conducted in [22]. Despite recent progress, the accuracy of the clustering requires improvement. A multi-dimensional information fusion method based on the Dempster–Shafer evidence theory was proposed in [23], which obtained a higher diagnosis accuracy of alarm sequences. The results showed that the diagnosis accuracy was affected by the quality of recorded fault labels in maintenance records. A weighted Hamming distance was proposed and applied in the similarity analysis of alarm lists to identify the fault category [24]. It did not require a time-consuming training procedure and was easy to apply. However, the improvement in accuracy is limited by the number of labeled alarm sequences.

The above research status shows that when diagnosing the wind turbine faults using triggered alarms, the few fault samples and low-quality fault labels have limited the

improvement of diagnosis accuracy. Both factors are related to maintenance records because the fault types that trigger alarms are recorded in maintenance records. However, due to the self-inspection function of wind turbines and the irregular work of the operator, a large proportion of alarms has no corresponding maintenance record. That is to say, a large proportion of alarms have no recorded fault labels. Therefore, the actual alarm data contain few labeled alarms and many unlabeled alarms. The existing studies mainly focus on the analysis of labeled alarms. As far as we know, there is no research about how to improve diagnosis accuracy with few labeled and many unlabeled alarms. In addition, the existing literature does not delve deeply into the relationship between individual alarms. Some studies only consider the temporal order or occurrence probability of individual alarms [15,21], while others only focus on the relationship between one alarm sequence and another [20,22,24], without considering individual alarms.

To fill this gap, this paper proposes a new fault diagnosis method for wind turbines with alarms. The proposed diagnosis method is designed based on the word embedding technique and a Siamese neural network. Firstly, the Skip-gram model in word embedding is employed to convert non-numerical alarm codes into real-valued vector representations, considering their sequential relationships and frequencies within the alarm sequence (the Skip-gram model will be described in detail in Section 3.2). Additionally, the pretraining technique in word embedding is utilized to explore the relationships among individual alarms in unlabeled alarm data. Subsequently, by further optimizing the alarm vectors obtained from pretraining using labeled alarm data, the joint utilization of labeled and unlabeled data is achieved. Secondly, the designed fault type diagnostic model based on the Siamese neural network for unknown alarm sequences can delve into the similarity features among alarm sequences (the diagnostic model based on the Siamese neural network will be specifically described in Section 3.3) and produce similarity scores. In this study, the criterion used to diagnose the fault type of unknown alarm sequences is the similarity score between the unknown alarm sequence and known alarm sequences. Therefore, the overall strategy of the proposed method can be divided into two steps. First, a Siamese convolutional neural network with an embedding layer (S-ECNN) model is proposed to distinguish different alarm sequences. Secondly, the fault category of an unknown alarm sequence is deduced by the similarity score obtained through the S-ECNN model.

The main contributions of this paper can be summarized as follows:

- 1. The unlabeled and labeled alarms can be collaboratively applied in the proposed S-ECNN model, which can effectively improve the fault diagnosis accuracy of wind turbines.
- 2. The potential relationships among individual alarms are captured in *n*-dimensional space using a word embedding method, which considers not only the alarm order but also the frequency of occurrence.

The rest of the paper is organized as follows: Section 2 describes the background of wind turbine alarms and maintenance, Section 3 presents the proposed fault diagnosis method, the results of experimental verification and discussions are provided in Section 4, and conclusions are presented in Section 5.

2. Background

In this section, a brief description of wind turbine alarms and maintenance records is given. Moreover, we analyze the control principle of a wind turbine's main control system when it deals with alarms, which will explain why there are many unlabeled alarms and few labeled alarms.

2.1. Wind Turbine Alarms

Wind alarm systems vary widely between manufacturers but generally share the same broad functionality. They monitor wind turbines' operational variables and trigger alarms when the signals exceed threshold limits. A sample of an alarm list is shown in Table 1. Alarms are recorded continuously in chronological order. The alarm records contain turbine number, triggering time, alarm types, alarm codes, alarm flags, and description. Among

them, the alarm code is the unique code of an alarm. The alarm flag represents the start or the end of each alarm. Hence, each alarm has two records.

Turbine Number	Triggering Time	Alarm Types	Alarm Codes	Alarm Flags	Description
P01	2017/5/22 16:30:05	Information	I2	Start	The wind turbine is started
P01	2017/5/22 17:38:18	Warning	A264	Start	The first measuring point temperature of generator stator is high
P01	2017/5/22 17:38:37	Warning	A264	End	The first measuring point temperature of generator stator is high
P01	2017/5/22 17:38:51	Fault	T21	Start	The communication of the pitch system is an error
P01	2017/5/22 17:38:52	Information	I2	End	The wind turbine is started
P01	2017/5/23 00:15:20	Fault	T21	End	The communication of the pitch system is an error

Table 1. A sample of an alarm list.

When a wind turbine experiences a fault, it can result in alterations to multiple variable values and the subsequent generation of multiple alarms. Nevertheless, these alarms, occurring in a short time frame, are not indicative of the specific fault type. As such, further analysis of the alarms is necessary to identify the underlying cause of the fault.

Furthermore, it can be observed that the alarm data are in non-numerical form. To efficiently analyze and process this data, it is necessary to convert these non-numerical data into numerical form. Finding a reasonable and effective transformation method is one of the problems addressed in this paper.

2.2. Maintenance Records

After a wind turbine stops due to alarms, manual inspections are arranged by maintenance personnel. The technicians investigate the turbine malfunction and document the specific details in maintenance records. Consequently, the fault type or tag that triggers the alarm is recorded in the maintenance records. Table 2 provides an example of a maintenance record. The record contains the turbine number, the start time and end time of maintenance activity, the actual faults, and the solutions to faults. However, not all faults can be found in the maintenance records. This is primarily because the wind turbine's main control system automatically handles certain alarms.

Table 2. Example of a record in the maintenance records.

Turbine Number	Start Time	End Time	Actual Faults	Solutions
P01	2016/12/21 17:34:00	2016/12/25 12:45:00	A slip ring is damaged	Replace the slip ring

To ensure the safety of wind turbine operation, the main control system responds to specific faults that trigger multiple alarms by performing different operations to eliminate them. The controlling principle is illustrated in Figure 1, wherein each alarm level corresponds to a particular severity of abnormality. When the alarm level is low, no operation is performed. When the alarm level is moderate, the wind turbine is restarted or reset. If the moderate-level alarm persists even after a restart or reset, the wind turbine is shut down. When the alarm level is high, the wind turbine is immediately shut down. After the shutdown, the main control system executes pre-set actions through the self-inspection function. If the alarms persist, manual maintenance is performed and the fault events are documented in the maintenance records.



Figure 1. The flowchart of control principal when a wind turbine deals with alarms.

From the above, we can draw the following conclusions:

- When a wind turbine is shut down due to alarms, manual maintenance will be performed. However, many alarms cannot cause a shutdown. Thus, the fault events that trigger these alarms are not available.
- Some alarms that can cause a shutdown are eliminated by the self-inspection function and thus have no recorded fault events.

In addition, during the actual maintenance activities, due to the irregular work of the operator, some maintenance details are missing. Thus, more alarms have no available fault events. In this paper, we name these alarms the unlabeled data. On the contrary, the alarms that have available fault events are named the labeled data. The fewer labeled data make it harder to diagnose wind turbine faults. On the other hand, the unlabeled data are generally ignored by the existing studies. We will address both issues in this paper.

3. The Proposed Fault Diagnosis Methodology

This paper proposes a novel fault diagnosis method for a wind turbine with alarms mainly based on the proposed S-ECNN model. The flow chart of the proposed methodology is shown in Figure 2. It can be divided into four phases: alarm data preprocessing, pretraining alarm vectors using the unlabeled data, training the proposed S-ECNN model using the labeled data, and fault diagnosis of the unknown alarm sequences based on similarity calculation. Specifically, the designed S-ECNN model is used to distinguish different alarm sequences based on the distance between the extracted discriminative features of input samples. The fault category of an unknown alarm sequence is deduced by the maximum average similarity score between it and the known alarm sequences.



Figure 2. The flow chart of the proposed fault diagnosis methodology.

3.1. Alarm Data Preprocessing

3.1.1. Segmenting Alarm Sequences

As mentioned above, one fault event of a wind turbine will trigger several alarms. These alarms are recorded continuously without distinguishing which fault event they belong to. First, the alarms that belong to the same fault events need to be selected.

In an alarm system, information alarms are generally to communicate changes in certain operating conditions. I2 is an information alarm, which indicates the changes in the wind turbine's operational condition. As shown in Figure 1, when a wind turbine is started from a shutdown, the alarm I2 with the flag of start is sent out. When a wind turbine is shut down from running, the alarm I2 with the flag of the end is sent out. Therefore, the alarms related to a fault event are generated between the start of I2 and the end of I2. Accordingly, the continuous alarms are segmented into alarm sequences using I2. The alarms in one obtained alarm sequence belong to the same fault events.

3.1.2. Removing Redundant Alarms

First, the repeated alarm records are removed. As mentioned above, an alarm has a unique code and two flags: start and end. Hence, although one alarm only happens once, it has two records in one alarm sequence. We merge the repeated alarm records and only remain the record of the start. Second, the chattering alarms are removed. Chattering alarms repeat with a high frequency within a short period and constitute the most common family of nuisance alarms. An alarm that is activated three or more times within one minute is often considered as belonging to the class of worst chattering alarms [25]. The chattering alarms are often redundant. Consequently, alarms that repeat three or more times in one minute are merged into one.

3.1.3. Building Dataset

Some alarm sequences have their fault events recorded in maintenance records, while others do not. In this part, we will match the alarm sequences with their maintenance records and build the labeled alarm sequence dataset. The alarm sequences without fault events will constitute the unlabeled alarm sequence dataset.

We use the end time of an alarm sequence and the start time of a maintenance record to match the alarm sequence and its maintenance record. The schematic diagram of the match criterion is shown in Figure 3. The end time of the *i*-th alarm sequence is expressed as t_i^{end} , and the start time of the *j*-th subsequent maintenance record is expressed as t_i^{start} .

 t_i^{end} should be earlier than t_j^{start} . The alarm sequence corresponding to the maintenance activity is the last one. Thus, the alarm sequences that matched with maintenance records constitute the labeled alarm sequence dataset, which is expressed as Dataset A. The other alarm sequences constitute the unlabeled alarm sequence dataset, expressed as Dataset B.



Figure 3. The match criterion of alarm sequences and maintenance records.

3.2. Pretraining Alarm Vectors

The unlabeled alarm sequences are often ignored and not fully utilized in the existing methods. However, they are generated by the alarm system under a normal alarm mechanism and thus contain potential information about the relationship among individual alarms. This effective information can help in the fault diagnosis of alarm sequences. In this paper, a word embedding method is used to mine the potential relationship between individual alarms.

In natural language processing (NLP), words of plain text can be transformed into real-valued data by the word embedding method [26]. The words are represented as vectors so that machine learning algorithms can be used in various NLP tasks. In a comparison study of various word embedding methods, Naili et al. [27] concluded that Word2Vec worked better for word representation within a low-dimensional semantic space. Word2Vec is a neural-network-based word embedding method, which includes two models: the continuous bag-of-words (CBOW) and the Skip-gram model. The Skip-gram model has several advantages compared to CBOW: (1) Flexibility: The Skip-gram model is more flexible as it predicts the context words given a target word. This allows it to capture a wider range of contextual information, resulting in a better representation of word semantics. (2) Handling rare words: The Skip-gram model performs better in handling rare words (low-frequency words). Unlike CBOW, which sums up the vectors of context words, Skipgram avoids the dominance of dense high-frequency words, enabling better capturing of rare word features. (3) Modeling short texts: The Skip-gram model performs better when dealing with short texts. CBOW, due to the summing operation on context word vectors, may lose some contextual information in short texts. Skip-gram, by predicting each context word individually, better preserves the semantic information in short texts. Accordingly, the Skip-gram model is used in this study.

The Skip-gram model [28] regards a corpus of words as inputs and produces a corresponding vector from R^n (*n* is the embedding-space dimension) for each unique word in the corpus. In the embedding space, the vectors of words that occur regularly nearby in the corpus are positioned close. Therefore, the word vectors capture and express the contextual similarities of words. When training the Skip-gram model using unlabeled alarm sequences, the single alarm is regarded as a word expressing semantics. An alarm sequence is regarded as a sentence describing a fault of a wind turbine. All the unlabeled alarm sequences constitute a corpus.

A brief description of the Skip-gram is given as follows [29]:

Given a sentence $s = w_1, w_2, ..., w_i, ..., w_n(w_i \in D)$, D is the collection of words. We model each word w_i by using its context words $w_{i-ws}, ..., w_{i-1}, w_{i+1}, ..., w_{i+ws}$, where $2 \times ws$ is the width of the considering context window. The center word and context words are projected into two types of embeddings v_i and $v'_{i+j}(1 \le |j| \le ws)$, respectively, as shown in Figure 4. The training goal of the Skip-gram model is to find word vector representations that help predict contextual words in a sentence. Given a training corpus with *N* sentences $C = \{s_c = w_1, w_2, \dots, w_{n_c}\}|_{c=1}^N$, the training objective is to minimize:

$$L_{SG} = -\sum_{c=1}^{N} \sum_{i=1}^{n_c} \sum_{1 \le |j| \le ws} \log f(v'_{i+j}, v_i),$$
(1)

Herein, $f(v'_{i+j}, v_i) = p(w_{i+j}|w_i)$ represents the concurrence probability of the word w_{i+j} when a word w_i is given, which is estimated by:

$$p(w_{i+j}|w_i) = \frac{\exp(v'_{i+j}^{\top}v_i)}{\sum_{w_k \in D} \exp(v'_k^{\top}v_i)}.$$
(2)

Eventually, alarms are represented as *n*-dimensional vectors, in which both the order of word occurrence and the frequency of occurrence are considered.



Figure 4. The Skip-gram model.

They are completely dependent on the relationship among distinct alarms in an alarm sequence without considering the fault events. In the next section, the alarm vectors will be further optimized using the labeled alarm sequences. Therefore, the obtained alarm vectors in this section are named the pretrained alarm vectors.

3.3. The Proposed S-ECNN Model

The proposed S-ECNN model is based on the Siamese neural network, which was proposed by Bromley et al. for one-shot learning, setting out to identify the similarities of signatures on cheques [30]. It has been widely used to leverage similarities of input sample pairs for many tasks (e.g., image recognition [31,32] and anomaly detection [33,34]). A Siamese neural network consists of two networks with the same structure and shared weights. The network reads two inputs, maps them to the target space respectively, and then uses a distance function to join them for similarity metric. The network of symmetric structure guarantees that two similar inputs will be mapped to similar feature space, while distinct inputs can be effectively differentiated.

The structure of the proposed S-ECNN model is shown in Figure 5. It comprises two identical one-dimensional deep convolutional neural networks, which are used to extract the discriminative features of inputs. The one-dimensional deep convolutional neural network consists of an embedding layer connected to a 1D-CNN. The embedding layer is at the beginning of the basic structure, which is fed a pair of alarm sequences. The embedding matrix obtained from the pretrained alarm vectors is used to initialize the parameters of the embedding layer. The difference between two discriminative features is computed in the distance layer. Eventually, a fully connected layer with a sigmoid activation function is used to give the probability of the label.



Figure 5. The structure of the proposed S-ECNN model.

3.3.1. The Embedding Layer

The embedding layer is a neural network, which can turn positive integers (indexes) into dense vectors of fixed size. It is usually placed at the beginning of a network to transform categorical non-numerical data into a categorical dense vector representation. Afterward, through downstream supervised learning, the categorical dense vectors can be continuously trained and optimized.

In this paper, we apply the embedding layer to transform an alarm in the form of a code into a vector in the form of real-valued data. Let $x_{i,k}$ be the *k*-th alarm code in the *i*-th alarm sequence. An alarm sequence of length *l* (padded where necessary) is represented as:

$$X_{i} = (x_{i,1}, x_{i,2}, \dots, x_{i,k}, \dots, x_{i,l}) (x_{i,k} \in E),$$
(3)

where $X_i \in A$ is a labeled alarm sequence, and E is the collection of alarm codes configured in the wind turbine alarm system. The transformed *n*-dimensional vector is expressed as $x_{i,k} \in \mathbb{R}^n$. Then, the alarm sequence can be represented as:

$$X_{i} = [x_{i,1}, x_{i,2}, \dots, x_{i,k}, \dots, x_{i,l}]^{1},$$
(4)

where $X_i \in \mathbb{R}^{n \times l}$ is the matrix representation of the *i*-th alarm sequence.

The initialization parameters of an embedding layer can be random or assigned by loading an embedding matrix. In this paper, we initialize the parameters by loading the embedding matrix obtained from the pretrained alarm vectors. After loading the embedding matrix into the embedding layer, the embedding vectors will be continuously updated in the training phase using the labeled alarm sequences. Thus, the labeled alarm sequences and unlabeled alarm sequences can be collaboratively applied.

3.3.2. 1D-CNN

The embedding layer is followed by a 1D-CNN. CNN is a type of feedforward neural network that is useful for processing data that have a degree of spatial correlation between local data points. In contrast with other network models, the parameters sharing property of convolution reduces the number of parameters to be optimized. Hence, the training

efficiency and the scalability of the model are improved. In this paper, we design a 1D-CNN to extract the discriminative feature of an alarm sequence. The structure of the network is listed in Table 3.

Fable 3. Structure of the 1D-CNI

Layers	Filters	Stride	Output Size	Layers
Convolutional-ReLU	128 filters size of 3×100	1	128 imes 30 imes 1	Convolutional-ReLU
Max-Pooling	3	3	128 imes 10 imes 1	Max-Pooling
Convolutional-ReLU	32 filters size of 3	1	32 imes 10 imes 1	Convolutional-ReLU
Max-Pooling	3	3	32 imes 4 imes 1	Max-Pooling
Flatten layer	-	-	128×1	Flatten layer

It contains two convolutional layers with filters of varying sizes. Each convolutional layer is followed by a max-pooling layer. Afterward, feature maps are flattened into a single vector. The vector is the discriminative feature of the input alarm sequence and is expressed as:

$$\mathbf{i}_i = f(\mathbf{X}_i),\tag{5}$$

where $h_i \in R^{m \times 1}$, $X_i \in R^{n \times l}$ is the matrix of the *i*-th alarm sequence, and $f(\cdot)$ denotes the feature vector extraction process of the proposed 1D-CNN.

ł

3.3.3. Distance Layer and Output Layer

Two 1D-CNNs are merged in the distance layer. The distance between two discriminative features is calculated based on the pairwise Euclidean distance. Suppose $(X_j^1, X_j^2, y_j), (j = 1, 2, ..., M)$ be a pair of inputs that is randomly selected from the collection A, where $y_j = 0$ if two inputs belong to the same fault category, otherwise, $y_j = 1$. *M* is the total number of input pairs. The discriminative feature vectors of each input are expressed as h_i^2 and h_i^2 , respectively. The Euclidean distance can be denoted by:

$$D_j = \left| \boldsymbol{h}_j^1 - \boldsymbol{h}_j^2 \right|. \tag{6}$$

After that, a fully connected layer with a sigmoid activation function is followed. The neurons in the fully connected layers are dropped out with a probability of 0.3. This neuron computes the prediction of the input pair as:

$$p(X_j^1, X_j^2) = \sigma(\sum_j \alpha_j \left| \boldsymbol{h}_j^1 - \boldsymbol{h}_j^2 \right|), \tag{7}$$

where $\sigma(\cdot)$ is the sigmoid non-linearity function, and α_j is a learnable parameter representing the importance of D_j . The output $p(X_j^1, X_j^2)$ is between zero and one, which scores the probability of the label. At the same time, the output $p(X_j^1, X_j^2)$ is the normalized representation of the difference between inputs. The normalized representation is transformed by the sigmoid non-linearity function mentioned above. The value of the output $p(X_j^1, X_j^2)$ is between zero and one. The bigger the value of $p(X_j^1, X_j^2)$ is, the bigger the difference is. Let $S(X_i^1, X_i^2)$ be the similarity between two inputs. Thus, it is calculated as:

$$S(X_i^1, X_j^2) = 1 - p(X_j^1, X_j^2).$$
(8)

The value of the similarity is also between zero and one. The bigger the value of $S(X_j^1, X_j^2)$ is, the bigger the similarity is.

The binary cross-entropy function is used as the loss function. It aims to minimize the distance between samples of the same category while maximizing the distance between samples of different categories. The loss function has the following form:

$$Loss = \sum_{j=1}^{M} y_j \log(p(X_j^1, X_j^2)) + (1 - y_j) \log(1 - p(X_j^1, X_j^2)).$$
(9)

3.4. Fault Diagnosis of Unknown Alarm Sequences

As described in Section 3.3.3 above, we can obtain the similarity score between two alarm sequences through the proposed S-ECNN model. When predicting the fault of one unknown alarm sequence, we compare it with every labeled alarm sequence and obtain the similarity score of each pair. The label with the maximum average similarity score will mark the unknown alarm sequence. Suppose the similarity score between one unknown alarm sequence X'_p with a fault label μ^q is:

$$S(X', X_p^q), (p = 1, 2, \dots, P^q, q = 1, 2, \dots, Q),$$
 (10)

where *Q* is the total number of fault categories, and P^q is the total number of alarm sequences with a label μ^q . The calculation method of the similarity score is the same as Equation (8). The average similarity score \overline{S} between X' and the alarm sequence X^q with a label μ^q is calculated as:

$$\overline{S}(X', X^q) = \frac{1}{P^q} \sum_{p=1}^{P^q} S(X', X^q_p).$$
(11)

The maximum average similarity score is expressed as:

$$\overline{S}_{\max}(X', X^f) = \max\left\{\overline{S}(X', X^1), \dots, \overline{S}(X', X^Q)\right\}.$$
(12)

Accordingly, the label of the unknown alarm sequence is μ^{f} . In other words, the fault type of the unknown alarm sequence is diagnosed as fault *f*.

4. Results and Discussion

4.1. Data Description

The data used in this paper are from a wind farm located in China. There are 24 wind turbines on the wind farm, installed with direct-drive, variable-speed, and variable-pitch generators. The available alarm data and maintenance records are from May 2016 to October 2017. There are a total of 261 maintenance records.

First, alarm data were preprocessed. After segmenting alarm sequences using the alarm I2, we obtained 1626 alarm sequences. An alarm sequence is given as an example and shown in Figure 6. To maintain confidentiality, the turbine number and the description have been concealed. The raw alarm sequence contains 31 alarms. The blue-font alarms are repeated alarm records. The red-font alarms are chattering alarms. After removing redundant alarms, there are still 22 alarms remaining in the alarm sequence. Afterward, the labeled alarm sequence dataset (Dataset A) and the unlabeled alarm sequence dataset (Dataset B) were built by matching the alarm sequences and maintenance records. As shown in Figure 6, the fault type that triggered the example alarm sequence is pitch motor driver failure and labeled as fault y_0 . Later, by loading pretrained alarm vectors, an individual alarm is transformed into an *n*-dimensional vector representation, and the alarm sequence is paired with another alarm sequence as an input pair, which is fed into the S-ECNN model.



The alarm sequence after data preprocessing

Figure 6. Data preprocessing of an alarm sequence. The blue-font alarms are repeated alarm records. The red-font alarms are chattering alarms.

For all the data, we obtained 261 labeled alarm sequences. For the sake of verification, we selected the fault categories that occurred more than six times to form Dataset A. The final Dataset A consists of 74 alarm sequences, and Dataset B consists of 1365 alarm sequences. For Dataset A, the fault categories and the number of each category are listed in Table 4. About 75% of the alarm sequences for each fault category form the training set, and the others form the test set.

Table 4. The fault categories and the number of labeled alarm sequences.

Label	Fault Categories	Number of Alarm Sequences (Training Set/Test Set)
F1	Hub speed encoder fault	8 (6/2)
F2	Pitch system communication fault	8 (6/2)
F3	Vibration sensor fault	9 (7/2)
F4	Pitch motor driver fault	10 (8/2)
F5	Generator stator fault	10 (8/2)
F6	Frequency-converter communication fault	14 (11/3)
F7	Wind vane fault	15 (11/4)

For the proposed S-ECNN model, the input is a pair of alarm sequences, which are selected from Dataset A. Suppose there are M alarm sequences in the training set. The number of non-repeated combinations of choosing two alarm sequences from Dataset A to form an input pair is calculated as:

$$C_M^2 = 0.5M(M-1).$$
 (13)

Therefore, the number of constructed input pairs is 1596. Among them, 217 pairs of inputs are with the same fault category, and 1379 pairs of inputs are with different fault categories. Thus, the number of training samples is remarkably enlarged by constructing input pairs. The input pairs for the test set are constructed in the same way.

4.2. Model Variants

To demonstrate the advantages of collaboratively using the labeled and unlabeled alarm sequences, two variants of the proposed S-ECNN model are given. They are the S-ECNN-rand model and the S-ECNN-static model. In the S-ECNN-rand model, the pretrained alarm vectors are not applied in the parameter initialization of the embedding layer. The parameters are randomly initialized. Therefore, the diagnosis results of the S-ECNN-rand model are only based on the labeled alarm sequences. In the S-ECNN-static model, the pretrained alarm vectors are applied in the parameter initialization of the embedding layer. When training the model, the parameters of the embedding layer are kept static. In other words, the vector representations of alarms are determined by the unlabeled alarm sequences and not updated in the embedding layer during the training process. The gradients are backpropagated to the first convolutional layer. Only the parameters of networks after the embedding layer are learned using the labeled alarm sequences. The training process can be regarded as a downstream classification task to evaluate the pretrained alarm vectors. Therefore, the diagnosis results of the S-ECNN-static model are mainly based on the unlabeled alarm sequences.

4.3. Evaluation of Pretrained Alarm Vectors

Obtaining word vectors for a domain-specific corpus requires fine-tuning of model parameters. The key parameters of the Skip-gram model are word vector dimension *d* and window size *w*. To adjust the model parameters, the evaluation of the obtained alarm vectors is needed. Word vector training is an unsupervised process. When evaluating the obtained vectors, word embedding can be used as input features to a downstream task and measure changes in performance metrics specific to that task. As mentioned above, the S-ECNN-static model can be regarded as a downstream classification task of alarm vector embedding, so it is used to fine-tune the parameters of the Skip-gram model. The accuracy is used to evaluate the performance of classification, which is defined as:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN'}$$
(14)

where *TP* represents the number of positive instances, which are classified correctly; *FP* represents the number of negative instances, which are misclassified; *TN* represents the number of negative instances, which are classified correctly; *FN* represents the number of positive instances, which are misclassified [35].

The accuracies of the S-ECNN-static model under different word vector dimensions and window sizes are shown in Figure 7. When the word vector dimension d is reduced from 100 to 75, 50, 25, and 10, the accuracy decreases. However, increasing the dimension w from 100 to 150 and 200, the accuracy does not increase significantly. When the word vector dimension d is 100 and the window size w is 4, the accuracy is the highest, and the pretrained alarm vectors are adopted in this paper.



Figure 7. The accuracies of the S-ECNN-static model with different parameters.

To visualize the alarm vectors, we reduced the model dimension to a three-dimensional space using t-distributed stochastic neighbor embedding (t-SNE) [36], which is known to be one of the very powerful tools for dimensionality reduction and data visualization. The visual display of alarm vectors is shown in Figure 8. Three types of pitch system alarms are highlighted as examples. As we can see, the pitch system alarms of the same type show a tendency for aggregation. The more complex relationship between alarms cannot be reflected in the three-dimensional visual image. Further algorithm analysis is required.



Figure 8. Visualization of alarm vectors in the three-dimensional space.

4.4. Evaluation of Experimental Results

Experiments were conducted to determine the optimal hyperparameters of the proposed S-ECNN model, which was built with Tensorflow 1.15.4 in Python based on Anaconda Spyder. The training set was further divided into two subsets, where one was for the training, and the other served for the validation. We optimized the structure parameters of 1D-CNN and the optimizer of the whole model, and the optimal structure parameters have been listed in Table 3. The model's accuracy curves and loss curves for the training set and validation set are shown in Figure 9. As the epoch increases, the training accuracy has been improved, but the validation accuracy increases first and then decreases. The training loss decreases, but the validation loss decreases first and then increases. It shows that with the increase of epochs, the model has over-fitting. Therefore, we retain the model when the epoch is 21 before over-fitting.



Figure 9. The accuracy curves and loss curves for the training and validation set of the proposed method.

4.4.1. Evaluation of Distinguishing Ability

The distinguishing ability of the proposed S-ECNN model was compared with that of its variants. The confusion matrix of binary classification was used to analyze the comparing results, as shown in Figure 10. Each column of the matrix represents the instances in a

predicted label while each row represents the instances in an actual label. The effectiveness was further quantified by accuracy and the following widely used indicators [35]:

$$Recall = \frac{TP}{TP + FN},$$
(15)

$$Precision = \frac{TP}{TP + FP'},\tag{16}$$

$$Specificity = \frac{TN}{TN + FP},\tag{17}$$

$$F1 - score = \frac{2 \cdot Recall \cdot Precision}{Recall + Precision}.$$
(18)



Figure 10. The confusion matrices for distinguishing ability evaluation.

The comparing results of the proposed S-ECNN model and its variants are listed in Table 5. The results indicate that the proposed S-ECNN model is more effective than the S-ECNN-rand model and the S-ECNN-static model because all the indicators of the S-ECNN model are the highest. Specifically, our concern is whether the model can accurately identify the same fault category and whether the model can effectively distinguish different fault categories. The model is not directly used for prediction. In this section, recall is calculated as the number of correct identifications of the same fault category divided by the number of given same-fault sample pairs, indicating the ability to identify the same fault category. Specificity is calculated as the number of correct identifications of the different fault categories divided by the number of given different-fault sample pairs, indicating the ability to distinguish different alarm sequences. The S-ECNN achieves a recall of 89.5% and a specificity of 86.3%, both of which are satisfied. Owing to the imbalanced data, precision is not good. However, in this section, *precision* is calculated as the number of correct identifications of the same fault category divided by the number of sample pairs predicted as the same fault category, indicating the ability for prediction. Therefore, compared with precision, we pay more attention to recall. Additionally, F1-score is also given considering imbalanced classification. Compared with the S-ECNN-rand model, the F1-score of the S-ECNN model increases by 12.9%. Therefore, the collaborative use of the labeled and the unlabeled alarm sequences can effectively improve the distinguishing ability.

Table 5. The comparing indicators for distinguishing different alarm sequences.

Model	Accuracy	Recall	Precision	Specificity	F1-Score
S-ECNN-rand	78.7%	84.2%	38.1%	77.8%	52.5%
S-ECNN-static	79.4%	73.7%	37.8%	80.3%	50.0%
S-ECNN	86.8%	89.5%	51.5%	86.3%	65.4%

4.4.2. Evaluation of Fault Diagnosis Method

First, we compared the performance of the proposed method with that of its variants. The confusion matrix of the multi-class problem was used, as shown in Figure 11. Each column of the matrix represents the instances in a predicted fault category, while each row represents the instances in an actual fault category. The effectiveness of diagnosing each type of fault was further quantified by the mentioned indicators: *accuracy, recall, precision, specificity*, and *F1-score*. In this section, *recall* corresponds to the ability to identify a type of specific fault, *precision* represents the degree of success of methods when a fault type is predicted, and *specificity* corresponds to the capacity of methods to refuse the identification of a specific fault type.



Figure 11. The confusion matrices for fault category prediction result.

After obtaining the above indicators for each fault type, the overall classification performance was evaluated using the macro-average value of each indicator, which is calculated as:

$$Ave-indicator = \frac{1}{Q}\sum_{i=1}^{Q}indicator_{Fi},$$
(19)

where *Q* is the number of fault categories, Fi is the label of the fault. In this paper, Q = 7.

Table 6 shows the comparing results, which indicate that the proposed S-ECNN model is more effective than its variants in fault diagnosis. By collaboratively using the labeled and the unlabeled alarm sequences, all the indicators have been improved. Particularly, compared with the S-ECNN-rand model, the *ave-precision* of the S-ECNN model increases by 13.1%, indicating that this method is more effective at fault diagnosing while avoiding false identification.

Table 6. The comparing indicators with the model's variants for fault diagnosis.

Method	Ave-Accuracy	Ave-Recall	Ave-Precision	Ave-Specificity	Ave-F1_Score
S-ECNN-rand	93.3%	75.0%	77.4%	96.0%	76.2%
S-ECNN-static	91.6%	73.8%	76.2%	95.2%	75.0%
S-ECNN	97.0%	89.3%	90.5%	98.3%	89.9%

Second, we compared the proposed method with a cluster analysis (CA) method [22], a multi-dimensional information processing (MIP) method [23], and a similarity analysis (SN) method [24]. The alarm data were preprocessed and the alarm sequences were segmented using the proposed method in this paper. In the CA method, the density-based spatial clustering of applications with noise was applied. In the MIP method, each alarm sequence was labeled with the most possible fault. When using the SN method, about 75% of the alarm sequences for each fault category were used to extract the feature vectors. It is noteworthy that these methods can only analyze the labeled alarm sequences, while the proposed method collaboratively uses the labeled and the unlabeled alarm sequences. For a fair comparison, the S-ECNN-rand model, which only uses the labeled alarm sequences, is also involved in the following comparison.

The mentioned macro-average indicators were used to quantify the effectiveness of these methods. For the SN method, the fault type of one unknown alarm sequence may not be assigned with a historical fault. In this situation, the FP in a confusion matrix cannot be deduced. Then the *ave-precision*, *ave-specificity*, and *ave-F1-score* cannot be calculated. The comparing results are shown in Figure 12. As we can see, compared with the existing methods, the S-ECNN-rand model has achieved some success. The *ave-accuracy*, *ave*precision, ave-specificity, and ave-F1-score of the S-ECNN-rand model are higher than that of the existing methods, which proves the effectiveness for fault diagnosis of alarm sequences. However, the *ave-recall* of the S-ECNN-rand model is lower than that of the SN method, which indicates that the ability to identify a type of specific case for the S-ECNN-rand model is not good enough. For the proposed method, all the indicators have been further improved. Specifically, the proposed method achieves the highest *ave-accuracy* of 97%. In addition, compared with the SN method and the S-ECNN-rand model, the *ave-recall* of the proposed method increased by 8.1% and 14.3% respectively, indicating that this method could effectively identify a specific fault type. Meanwhile, the proposed method significantly improves the ave-precision and ave-F1-score compared with the S-ECNN-rand model. Finally, although the *ave-specificity* of the other methods can achieve some success, the value of the proposed method is still the highest.



Figure 12. The comparing indicators with existing methods for fault diagnosis.

5. Conclusions

This paper proposed a novel fault diagnosis method for wind turbines with alarms based on word embedding and a Siamese convolutional neural network. To improve diagnosis accuracy, the proposed method collaboratively used labeled alarm sequences and unlabeled alarm sequences. For the unlabeled alarm sequences, the potential relationship among alarms was mined using the Skip-gram model, and *n*-dimensional pretrained alarm vectors were obtained. For the labeled alarm sequences, the discriminative features were extracted to distinguish different alarm sequences by the proposed S-ECNN model, in which the pretrained alarm vectors were optimized and trained. The effectiveness of the proposed method was proved by using the actual alarm data of a wind farm in China. The accuracy of the proposed S-ECNN model for distinguishing different alarm sequences was 86.8%, which was higher than its variants. The result indicated that the collaborative use of the labeled and the unlabeled alarm sequences could effectively improve the distinguishing ability. The macro-average accuracy of the proposed method for fault diagnosing was 97.0%, which was higher than its variants and the existing three methods. The result indicated that the proposed method could effectively improve fault diagnosis accuracy. In addition, the embedding layer introduced in the proposed network provides the possibility of transfer learning, which will be further researched in the following works.

The method proposed in this paper utilizes word embedding to convert alarms into numerical vector representations. Furthermore, alarm sequences consisting of multiple alarms can also be represented in matrix form. In industrial settings, alarm codes continue to increase and are not presented in the form of alarm sequences. Therefore, it is of great research value to investigate how to predict the next alarm code based on historical alarm sequences and thereby forecast the type of failure that wind turbines are likely to experience. Additionally, studying the relationship between the occurrence of alarms and wind turbine power and load is another worthwhile research question. Based on the numerical representation of alarm sequences, alternative time series data mining models such as Bi-LSTM [37] can be used to establish prediction models.

Author Contributions: Conceptualization, methodology, software, validation and writing—original draft preparation, L.W. (Lu Wei); methodology, software, validation, J.Q.; formal analysis, investigation, resources and funding acquisition, Z.Q.; writing—review and editing, F.L., L.W. (Liliang Wang), and H.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the National Natural Science Foundation of China (No. 61573046) and the Program for Changjiang Scholars and Innovative Research Team in University (No. IRT1203).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

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

References

- Lee, J.; Zhao, F. Global Wind Report 2022; Global Wind Energy Council: Brussels, Belgium, 2022; 158p. Available online: https://gwec.net/global-wind-report--2022/ (accessed on 4 April 2022).
- Ren, G.; Liu, J.; Wan, J.; Guo, Y.D.; Yu, D. Overview of wind power intermittency: Impacts, measurements, and mitigation solutions. *Appl. Energy* 2017, 204, 47–65. [CrossRef]
- 3. Spinato, F.; Tavner, P.J.; Bussel, G.L.; Koutoulakos, E. Reliability of wind turbine subassemblies. *IET Renew. Power Gener.* 2009, *3*, 387–401. [CrossRef]
- Yang, W.; Tavner, P.J.; Crabtree, C.J.; Feng, Y.; Qiu, Y. Wind turbine condition monitoring: Technical and commercial challenges. Wind Energy 2014, 17, 673–693. [CrossRef]
- 5. Martin, R.; Lazakis, I.; Barbouchi, S.; Johanning, L. Sensitivity analysis of offshore wind farm operation and maintenance cost and availability. *Renew. Energy* **2016**, *85*, 1226–1236. [CrossRef]
- 6. Helbing, G.; Ritter, M. Deep Learning for fault detection in wind turbines. *Renew. Sustain. Energy Rev.* 2018, 98, 189–198. [CrossRef]
- Peeters, C.; Guillaume, P.; Helsen, J. Vibration-based bearing fault detection for operations and maintenance cost reduction in wind energy. *Renew. Energy* 2018, 116, 74–87. [CrossRef]
- Zhang, L.; Zhang, H.; Cai, G. The multiclass fault diagnosis of wind turbine bearing based on multisource signal fusion and deep learning generative model. *IEEE Trans. Instrum. Meas.* 2022, 71, 3514212. [CrossRef]
- 9. Tiboni, M.; Remino, C.; Bussola, R.; Amici, C. A review on vibration-based condition monitoring of rotating machinery. *Appl. Sci.* **2022**, *12*, 972. [CrossRef]
- 10. Coronado, D.; Wenske, J. Monitoring the oil of wind-turbine gearboxes: Main degradation indicators and detection methods. *Machines* **2018**, *6*, 25. [CrossRef]
- Shahriar, M.R.; Borghesani, P.; Tan, A.C. Electrical signature analysis-based detection of external bearing faults in electromechanical drivetrains. *IEEE Trans. Industr. Electron.* 2018, 65, 5941–5950. [CrossRef]
- 12. Wei, L.; Qian, Z.; Zareipour, H. Wind turbine pitch system condition monitoring and fault detection based on optimized relevance vector machine regression. *IEEE Trans. Sustain. Energy* **2020**, *11*, 2326–2336. [CrossRef]
- Jin, X.; Xu, Z.; Qiao, W. Conditon monitoring of wind turbine generators using SCADA data analysis. *IEEE Trans. Sustain. Energy* 2021, 12, 202–210. [CrossRef]
- 14. Wen, W.; Liu, Y.; Sun, R.; Liu, Y. Research on anomaly detection of wind farm SCADA wind speed data. *Energies* **2022**, *15*, 5869. [CrossRef]
- 15. Qiu, Y.; Feng, Y.; Tavner, P.; Richardson, P.; Erdos, G.; Chen, B. Wind turbine SCADA alarm analysis for improving reliability. *Wind Energy* **2012**, *15*, 951–966. [CrossRef]

- 16. Wang, J.; Yang, F.; Chen, T.; Shah, S.L. An overview of industrial alarm systems: Main causes for alarm overloading, research status, and open problems. *IEEE Trans. Autom. Sci. Eng.* **2016**, *13*, 1045–1061. [CrossRef]
- 17. Qiao, W.; Lu, D. A survey on wind turbine condition monitoring and fault diagnosis—Part I: Components and subsystems. *IEEE Trans. Ind. Electron.* **2015**, *62*, 6536–6545. [CrossRef]
- Wang, J.; Li, H.; Huang, J.; Su, C. A data similarity based analysis to consequential alarms of industrial processes. J. Loss Prev. Process Ind. 2015, 35, 29–34. [CrossRef]
- Rodríguez-López, M.A.; López-González, L.M.; López-Ochoa, L.M. Development of indicators for the detection of equipment malfunctions and degradation estimation based on digital signals (alarms and events) from operation SCADA. *Renew. Energy* 2022, 18, 288–296. [CrossRef]
- 20. Chen, B.; Qiu, Y.N.; Feng, Y.; Tavner, P.J.; Song, W.W. Wind turbine SCADA alarm pattern recognition. In Proceedings of the IET Conference on Renewable Power Generation, Edinburgh, UK, 5–8 September 2011.
- Tong, C.; Guo, P. Data mining with improved Apriori algorithm on wind generator alarm data. In Proceedings of the 25th Chinese Control and Decision Conference, Guiyang, China, 25–27 May 2013.
- Leahy, K.; Gallagher, C.; O'Donovan, P.; O'Sullivan, D.T. Cluster analysis of wind turbine alarms for characterising and classifying stoppages. *IET Renew. Power Gener.* 2018, 12, 1146–1154. [CrossRef]
- Qiu, Y.; Feng, Y.; Infield, D. Fault diagnosis of wind turbine with SCADA alarms based multidimensional information processing method. *Renew. Energy* 2020, 145, 1923–1931. [CrossRef]
- 24. Wei, L.; Qian, Z.; Pei, Y.; Wang, J. Wind turbine fault diagnosis by the approach of SCADA alarms analysis. *Appl. Sci.* **2022**, *12*, 69. [CrossRef]
- 25. 20/30400796 DC; Management of Alarms Systems for the Process Industries. International Society of Automation: Miami, FL, USA, 2020.
- Zhang, C.; Guo, R.; Ma, X.; Kuai, X.; He, B. W-TextCNN: A TextCNN model with weighted word embeddings for Chinese address pattern classification. *Comput. Environ. Urban Syst.* 2022, 95, 101819. [CrossRef]
- 27. Naili, M.; Chaibi, A.H.; Ghezala, H.H.B. Comparative study of word embedding methods in topic segmentation. *Proc. Comput. Sci.* 2017, *112*, 340–349. [CrossRef]
- Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.; Dean, J. Distributed Representations of Words and Phrases and Their. Compositionality. Patent 10.48550/arXiv.1310.4546, 16 October 2013. Available online: https://arxiv.org/pdf/1310.4546.pdf (accessed on 16 October 2013).
- Cai, S.; Palazoglu, A.; Zhang, L.; Hu, J. Process alarm prediction using deep learning and word embedding methods. *ISA Trans.* 2019, 85, 274–283. [CrossRef] [PubMed]
- 30. Bromley, J.; Guyon, I.; LeCun, Y.; Säckinger, E.; Shah, R. Signature verification using a "siamese" time delay neural network. *Int. J. Pattern Recognit. Artif. Intell.* **1993**, *7*, 669–688. [CrossRef]
- Huang, L.; Chen, Y. Dual-path siamese CNN for hyperspectral image classification with limited training samples. *IEEE Geosci. Remote Sens. Lett.* 2021, 18, 518–522. [CrossRef]
- 32. Bharadwaj, S.; Prasad, S.; Almekkawy, M. An upgraded siamese neural network for motion tracking in ultrasound image sequences. *IEEE Trans. Ultrason. Ferroelectr. Freq. Control* 2021, *68*, 3515–3527. [CrossRef]
- Zhou, X.; Liang, W.; Shimizu, S.; Ma, J.; Jin, Q. Siamese neural network based few-shot learning for anomaly detection in industrial cyber-physical systems. *IEEE Trans. Ind. Inform.* 2021, 17, 5790–5798. [CrossRef]
- 34. Zhu, J.; Jang-Jaccard, J.; Watters, P.A. Multi-loss siamese neural network with batch normalization layer for malware detection. *IEEE Access* 2020, *8*, 171542–171550. [CrossRef]
- 35. Witten, I.H.; Frank, E.; Hall, M.A. Credibility: Evaluating what's been learned. In *Data Mining: Practical Machine Learning Tools and Techniques*, 3rd ed.; Morgan Kaufmann: Burlington, MA, USA, 2011; pp. 147–187.
- Pezzotti, N.; Thijssen, J.; Mordvintsev, A.; Hollt, T.; Van Lew, B.; Lelieveldt, B.P.; Eisemann, E.; Vilanova, A. GPGPU linear complexity t-SNE optimization. *IEEE Trans. Vis. Comput. Graph.* 2020, 26, 1172–1181. [CrossRef]
- Geibel, M.; Bangga, G. Data reduction and reconstruction of wind turbine wake employing data driven approaches. *Energies* 2022, 15, 3773. [CrossRef]

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