A Network Traffic Prediction Method for AIOps Based on TDA and Attention GRU

Kun Wang 1,2, Yuan Tan 1, Lizhong Zhang 2, Zhigang Chen 1,* and Jinghong Lei 1,©

1 School of computer science, Central South University, Changsha 410008, China
2 Information and Communication Company of State Grid Ningxia Electric Power Co., Ltd., Yinchuan 753000, China
* Correspondence: czg@csu.edu.cn

Abstract: Fault early warning is a challenge in the field of operation and maintenance. Considering the improvement of accuracy and real-time standards, as well as the explosive growth of operation and maintenance data, traditional manual experience and static threshold can no longer meet the production requirements. This research fully digs into the difficulties in fault early warning and provides targeted solutions in several aspects, such as difficulty in feature extraction, insufficient prediction accuracy, and difficulty in determining alarm threshold. The TCAG model proposed in this paper creatively combines the spatiotemporal characteristics and topological characteristics of specific time series data to apply to time series prediction and gives the recommended dynamic threshold interval for fault early warning according to the prediction value. A data comparison experiment of a core router of Ningxia Electric Power Co., Ltd. shows that the combination of topological data analysis (TDA) and convolutional neural network (CNN) enables the TCAG model to obtain superior feature extraction capability, and the support of the attention mechanism improves the prediction accuracy of the TCAG model compared to the benchmark models.

Keywords: time series prediction; failure warning; attention GRU; TDA; CNN; AIOps

1. Introduction

With the rapid development of machine learning and deep learning, researchers have shifted their attention from traditional automatic operation and maintenance to artificial intelligence for IT operations (AIOps). Major companies, including the state grid, Google, Microsoft, and Baidu, are deploying research in this field.

The term AIOps comes from Gartner [1], who aimed to enable software and service engineers to effectively use artificial intelligence and machine-learning technologies to efficiently build and operate services that are easy to support and maintain. AIOps is of great value, which can effectively ensure high service quality and customer satisfaction, improve engineering efficiency, and reduce operating costs. Its applications include anomaly detection [2–4], cluster analysis [5], fault prediction [6–8], and cost optimization [9].

Machine learning and deep learning are effective methods to solve classification and regression problems. However, when these methods are applied to AIOps, there are mainly the following challenges:

(1) With the gradual deployment of 5G base stations, the data acquisition capacity of various information equipment has been greatly improved, and the amount of operation and maintenance data has also increased explosively. How to extract the most concise and effective feature expression from a massive operation and maintenance data is a major challenge.

(2) Most of the operation and maintenance data exist in the form of time series. The traditional machine-learning model cannot learn the long-term correlation information, while the deep-learning models, such as DCNN and Bert, have problems, such as huge
super parameters, long training times, and the reliance on more GPU resources. Therefore, most of the current industrial application models are a compromise between model complexity, resource consumption, and time consumption.

3) The traditional automatic operation and maintenance mode is solidified and depends more on manual experience. At this stage, AIOps also needs to set rules manually in many scenarios. Therefore, how to use the algorithm model to automatically learn various rules to reduce manual intervention is not only the continuous goal of AIOps but is also an urgent problem to be solved.

Based on the above challenges, aiming at the problem of fault early warning in AIOps, this paper proposes a set of intelligent operation and maintenance architecture. The architecture can predict the value of the future period according to the historical data of information equipment and set the dynamic threshold interval based on predicted value to eliminate the dependence on manual experience and overcome the problem that the traditional threshold setting methods are difficult to deal with frequent scene changes. Specifically, the architecture first uses the convolutional neural network to extract spatiotemporal features in time series and then creatively applies topological data analysis to time series to extract topological features in data. Finally, the neural network model is trained using the extracted spatiotemporal and topological features to obtain the predicted value and dynamic threshold interval. To deal with the long-term dependence on time series data, the attention mechanism is applied in the gated cyclic unit (GRU) and is then used as the prediction model. The main contributions of this paper can be summarized as follows.

1) To solve the problem of fault early warning in AIOps, we propose a novel dynamic threshold-setting mechanism that calculates a threshold interval that automatically adjusts to scene changes based on prediction results and can fully consider the prediction error caused by the increase in step size.

2) Owing to the possible instability and nonlinearity in time series and long-time dependence, the performance of traditional feature extraction methods is not satisfactory. To obtain the most concise and effective feature expression of the data, this study uses a convolutional neural network to extract the spatio-temporal features in the data and uses topological data analysis to extract the topological features in the data. As far as the authors are aware, this is the first time topological data analysis has been applied to research in the field of AIOps.

3) To obtain more accurate prediction results and realize more intelligent fault early warning, this paper proposes the TCAG model, which connects the spatiotemporal and topological characteristics of data, trains the GRU neural network, and automatically calculates the dynamic threshold interval according to the prediction results. In addition, this study applied the attention mechanism to the output of the GRU at each time step. Experiments conducted on the core router data of State Grid Ningxia Electric Power Co., Ltd., show that the TCAG model is superior to the existing benchmark model in terms of prediction accuracy and robustness.

In the rest of this article, the second section refers to related work, and the third section proposes a system model for fault early warning. In the fourth section, this article introduces the modeling process and comparative experiments, the fifth section displays the experimental results and analysis, and the sixth section summarizes and prospects for this article.

2. Related Work

Previous research on intelligent operation and maintenance can be divided primarily into two categories: traditional machine-learning methods and deep-learning methods. Methods based on machine learning mainly include support vector machines, decision trees, random forests, and other models. Methods based on deep learning mainly include cyclic neural networks, convolutional neural networks, and residual neural networks.
2.1. Methods Based on Machine Learning

Jin et al. [10] proposed a single-index anomaly detection algorithm, which consists of an isolated forest (IF) algorithm, support vector machine (SVM) algorithm, local outlier factor (LOF) algorithm, and 3σ principle composition. The algorithm achieved an average score of 0.8304 for the sample data from the 2020 International AIOps Challenge. Soualhi, Medjaher, and Zerhouni [11] used an SVM algorithm for bearing fault diagnosis and an SVR algorithm for time series prediction to estimate the remaining service life of the bearing to monitor the health status of the key components of the bearing. Zhao et al. [12] proposed a general framework called “Period”. The framework clusters the daily sequence by using algorithms, such as ED, DBSCAN and K-means, to accurately detect the periodic distribution of the sequence. Finally, the performance of anomaly detection is improved by adapting to different periodic distributions. Since the performance of a single prediction model on unbalanced data sets is not adequate, Wang et al. [13] uses the prediction results calculated by three algorithms (XGBoost classification, LSTM classification, and XGBoost regression) as the feature input of stacked integrated-learning models that can generate more results to obtain robust prediction results. Experimental results show that the proposed stack ensemble-learning model can accurately predict disk failures 14 to 42 days in advance.

The machine-learning model can adapt to a small amount of data and relatively simple application scenarios. However, with the advent of the 5G era, all kinds of structured and unstructured data are growing explosively. Limited by the complexity of the model, the machine-learning model is gradually becoming stretched in the current application.

2.2. Method Based on Deep Learning

In recent years, with the proposal of residual neural network and gated cyclic network, the depth of neural network has been greatly expanded. Larger network layers and more network parameters enable the model to adapt to more complex scenarios; thus, deep learning begins to shine in various fields, including intelligent operation and maintenance.

Khalil et al. [14] proposed an early fault prediction method for circuits. This method uses fast Fourier transform (FFT) to obtain fault frequency characteristics, principal component analysis (PCA) to obtain dimensionality reduction data, and convolutional neural network (CNN) to learn and classify faults. Wen et al. [15] proposed a new CNN model based on LeNet-5 for fault diagnosis. By converting the signal into a two-dimensional (2D) image, this method can extract the features of the converted two-dimensional image and eliminate the influence of manual features.

In [16], a combination of a deep belief network, an autoencoder, and LSTM were introduced into the field of renewable energy power forecasting. Compared to the standard MLP and physical prediction models, the results of the deep-learning algorithm showed excellent prediction performance. In [17], a framework based on a long-term short-term memory (LSTM) recurrent neural network was proposed, which can adapt to the high volatility and uncertainty in time series to accurately predict the power load of a single user. Chen et al. [18] proposed a short-term power load forecasting model based on the deep residual network, which can integrate domain knowledge and researchers’ understandings of tasks with the help of different neural network building blocks. Several test cases and comparisons with existing models show that the proposed model provides accurate load forecasting results and has high generalization ability.

To summarize, the traditional machine-learning model is difficult to deal with unstructured data and adapt to massive big-data scenarios. The neural network model needs a deeper number of layers to obtain the full picture of the data, needs to consider the balance between fitting and receptive field, and the deep neural network model depends on more system resources. Therefore, considering the above problems, this paper proposes a TCAG combination model.
3. System Model

3.1. Topological Data Analysis (TDA)

TDA is an emerging field in complex data analysis, which studies some special geometric properties, which can remain unchanged after the shape of the graph is continuously changed and are labelled “topological properties” [19]. Through calculations using algebraic tools, the topological properties of the data in each spatial dimension can be strictly defined. For example, the two-dimensional space mainly includes the number of points and the degree of connection between the points, whereas the three-dimensional space mainly includes the number of hollow spheres and the degree of connection between the spheres.

3.1.1. Vietoris Rips Complex

TDA uses a simple complex method to express the original data shape. A simple complex is composed of one or more simplexes [20], which can simulate more complex shapes and is easier to deal with mathematically and computationally than the original graphics. A multidimensional simplex is shown in Figure 1.

![Simple complexes](image)

**Figure 1.** Simple complex diagram.

3.1.2. Continuous Homology

Continuous homology is a method used to measure the topological characteristics of shape and function. It can transform data into a simple complex and describe the spatial topology at different spatial resolutions [21, 22]. More persistent topologies can be detected on a wider spatial scale and are more representative of the real characteristics of the underlying space. The input of continuous homology is usually a point cloud or function [23], whereas the output depends on the nature of the analysis and is usually composed of a persistence graph or landscape.

The calculation steps of topological features are summarized as follows:

Step 1: Calculate the Euclidean distance matrix, \( D_E = \{ D_E(v_i, v_j) \} \), \( i_1, i_2 \in \{1, \ldots, N \} \).

Step 2: Construct the birth and death of homology groups for the added value of \( \lambda \).

For each \( \lambda \), use the closed loop \( B_3(v_i) \) of \( v_i \) with radius \( \lambda / 2 \) and \( K(\lambda) \) to calculate \( \tilde{\alpha}_{\tilde{p}, k} \).

If an older topology, \( \tilde{\alpha}_{\tilde{p}, k, 1} \), and a younger topology, \( \tilde{\alpha}_{\tilde{p}, k, 2} \), merge into a single \( \tilde{\alpha}_{\tilde{p}, k} \) at some point of \( \lambda \), \( \tilde{\alpha}_{\tilde{p}, k, 1} \) will become \( \tilde{\alpha}_{\tilde{p}, k} \) and \( \tilde{\alpha}_{\tilde{p}, k, 2} \) will die.

Step 3: The persistence graph is the output of a group of points representing the birth and death relationships of homology groups in the point cloud, expressed as

\[
\tilde{\Omega} = \left\{ \tilde{\alpha}_{\tilde{p}, k} = (\lambda_{\tilde{p}, k, 1}, \lambda_{\tilde{p}, k, 2}) : \tilde{p} = 0, 1, \ldots; k = 1, 2, \ldots, \tilde{p} \right\}
\]

Finally, \( \lambda_{\tilde{p}, k, 1} \) is drawn on the \( x \)-axis and \( \lambda_{\tilde{p}, k, 2} \) on the \( y \)-axis.

3.2. TCAG Model

In the existing models, TDA can extract stable and persistent topological features from data and transform the original data into a concise and effective feature expression; however, it may miss important information in the time dimension. CNN can extract the temporal and spatial characteristics of data and obtain important information on data in time and space dimensions, but it requires a deeper number of layers to present a more complete picture of the data. Therefore, in the TCAG model proposed in this study, the
advantages of the TDA and CNN were fully combined to express the topological and spatiotemporal features in the final feature set. Finally, the obtained feature set was used to train the attention GRU neural network, and an excellent prediction performance was obtained.

As shown in Figure 2, the TCAG model consisted of four modules. The original data first entered the dimensionality reduction module, and the features with high correlation were screened through correlation analysis and the Granger causality test to reduce the data dimension. The dimension-reduced data were entered into the topology data analysis module, and four types of topology features were obtained through continuous homology, including continuous landscape, continuous entropy, topology vector, and distance matrix. In addition, the dimensionality reduction data will also enter the CNN module to obtain the panorama of the data in the spatiotemporal dimension through multiple convolution and pooling operations. Finally, the extracted topological features were connected with spatiotemporal features to train the GRU neural network to obtain the final prediction results. In addition, this study applied an attention mechanism to the output of each time step of the GRU to filter out inefficiency information and obtain more accurate prediction results [24].

Figure 2. TCAG model diagram.

For the extraction of topological features, the time series data is first embedded into three-dimensional point cloud data through Takens embedding, and then, the Vietoris rips complex is constructed, and continuous homology is carried out. The result of persistent homology is a series of birth- and death-point pairs in the topological structure. Through the analysis and calculation of the point set, the persistent landscape, persistent entropy, and distance matrix can be obtained based on the Euclidean, bottleneck, and Wasserstein...
distances. The combination of these features is the final output of the module, which is a topological feature of the data.

The detailed design of the CNN module is shown in Figure 3, and the original data dimension are \((719 \times 6 \times n)\). The module uses a filter with step size of 1 and dimension of \((1 \times 3)\) and uses the “samepadding” to perform three convolution operations. The numbers of filters used are 16, 32, and 64, respectively. Using step size of 2 and dimension of \((1 \times 2)\) performs a maximum pool operation for the filter. Finally, the tensor is expanded by a flattened operation and connected to two full connection layers to obtain the final output of the CNN module—the spatiotemporal characteristics of the data.

![Figure 3. CNN model detailed structure diagram.](image)

To make better use of the extracted topological and spatio-temporal features and obtain more accurate prediction results, the attention GRU neural network model was used in this study. GRU is currently a popular time-series prediction model as it uses fewer parameters but can achieve the same prediction performance as LSTM in most scenarios [25,26]. The best prediction effect was obtained in this study.

To realize the early warning function of the TCAG framework, the dynamic threshold interval must be given according to the prediction results. Therefore, we designed a dynamic threshold setting mechanism, fully considering the influence of confidence and prediction step size on the threshold interval consulting about confidence intervals and naive forecast methods. Formula (2) shows the calculation of dynamic threshold interval.

\[
TR = \hat{y} \pm k \sqrt{\frac{1}{N} \sum_{i=1}^{N} (x_i - u)^2} \times \sqrt{n}
\]

(2)

where \(TR\) represents the dynamic threshold interval, \(\hat{y}\) represents the network traffic prediction result, \(N\) represents the sample size, \(x_i\) represents the i-th sample, \(u\) represents the average value of the sample, \(h\) is the prediction steps, and \(k\) represents the confidence factor, which is obtained from the selected confidence. Table 1 shows the relationship between confidence factor and value.

| Confidence | 80  | 85  | 90  | 95  | 96  | 97  | 98  | 99  |
|------------|-----|-----|-----|-----|-----|-----|-----|-----|
| Value      | 1.28| 1.44| 1.64| 1.96| 2.05| 2.17| 2.33| 2.58|
4. Experiment

4.1. Dataset Description

The dataset used in this experiment came from the monitoring data of a core router of State Grid Ningxia Electric Power Co., Ltd. with a time span of (1 May 2020 00:00–30 April 2021 23:55:00). The sampling frequency was five minutes, and 96,620 pieces of data were used. The prediction target of this experiment was the received traffic data of the core router.

4.2. Data Preprocessing

For the above datasets, this paper uses correlation analysis for feature selection, and the results are shown in Table 2.

Table 2. Characteristic correlation analysis.

|               | Send Traffic | Transmission Rate | Total Flow | Total Received Packets | Receiving Rate | Total Packets | Bandwidth Utilization | Receive Traffic |
|---------------|--------------|-------------------|------------|-------------------------|---------------|-----------------|----------------------|----------------|
| Memory usage  | 0.31         | 0.31              | 0.3        | 0.32                    | 0.29          | 0.32           | 0.32                 | 0.25            |
| Send traffic  | 1            | 1                 | 0.99       | 0.94                    | 0.97          | 0.94           | 0.94                 | 0.81            |
| Transmission rate | 1          | 1                 | 0.99       | 0.94                    | 0.97          | 0.94           | 0.94                 | 0.81            |
| Total packet error rate | −0.06     | −0.06             | −0.07      | −0.04                   | −0.08        | −0.04         | −0.04                | −0.08           |
| Total flow    | 0.99         | 0.99              | 1          | 0.94                    | 0.99          | 0.94           | 0.94                 | 0.83            |
| Total broadcast traffic | −0.06 | −0.06 | −0.07 | −0.04 | −0.08 | −0.04 | −0.04 | −0.08 |
| Total received packets | 0.94 | 0.94 | 0.94 | 1 | 0.94 | 1 | 1 | 0.8 |
| Receiving rate | 0.97         | 0.97              | 0.94       | 0.94                    | 0.94          | 0.94           | 0.94                 | 0.83            |
| Total packets | 0.94         | 0.94              | 0.94       | 1                       | 0.94          | 1              | 1                    | 0.8             |
| Total packets sent | −0.11 | −0.11 | −0.12 | −0.16 | −0.13 | −0.16 | −0.16 | −0.13 |
| Bandwidth utilization | 0.94 | 0.94 | 0.94 | 1 | 0.94 | 1 | 1 | 0.8 |
| Run time      | 0.22         | 0.22              | 0.21       | 0.28                    | 0.2           | 0.28           | 0.28                 | 0.2             |
| Receive traffic | 0.81         | 0.81              | 0.83       | 0.8                     | 0.83          | 0.8            | 0.8                  | 1               |

For topological data analysis, it was necessary to convert two-dimensional time-series data into three-dimensional point-cloud data. Therefore, after normalizing the data, the Takens embedding function in the gtda package was used for the conversion. The data before and after the conversion is shown in Figure 4. In 3-D point cloud map, points with different colors represent different homology groups.

![2-D time series data scatter diagram](image1)

![3-D point cloud data map](image2)

Figure 4. Data conversion diagram.

4.3. Evaluating Indicator

To accurately evaluate the experimental effect, the evaluation indexes used in this paper were $MSE, MAPE, R^2, 20\% ACC, 15\% ACC$, and $10\% ACC$.

$$MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$  \hspace{1cm} (3)
where $\hat{y}$ is the predicted value, $y$ is the actual value, $n$ is the number of samples, and $\text{count}(A > B)$ is the number of samples that meet the conditions of $A > B$.

5. Experimental Results and Analysis

To fully demonstrate the advantages of the TCAG model, three aspects are discussed in this section. First, the entire process of topology data analysis is shown in detail. Secondly, comparative experiments of different types of features were conducted, and finally, different models were carried out.

5.1. Topology Data Analysis

This experiment used topological data to analyze and mine hidden topological features in point cloud data. Through a continuous homology process, homologous topological structures were obtained at different scales. Figure 5 shows the homology group states four times during the continuous homology process.

In Figure 5, points with different colors represent different homology groups, and each subgraph is controlled by different super parameters. Among the four super parameters, $n_{\text{cubes}}$ is the resolution, representing the number of hypercubes in each dimension. The $n_{\text{cubes}}$ in chronological order is respectively 30, 15, 10, and 5. $n_{\text{nodes}}$ is the number of nodes, representing the number of topology. The $n_{\text{nodes}}$ in chronological order is respectively 600, 250, 125, and 25. In this experiment, the Euclidean distance was used as the measure of distance, and other super parameters include $\text{eps} = 0.1$, which represents the maximum distance that one of the two samples was considered to be near the other, and $\text{min} \_ \text{samples} = 15$, indicating the number of samples in the neighborhood where a point is regarded as a core point.

The upper-left figure shows the initial state for the first time. The distribution of homology groups is relatively discrete, and there is a spiral topology in the center, indicating that there is some periodicity in the original time series data. There are also many outliers around, which indicates that there is a large fluctuation in the original time-series data. The upper-right figure shows the state at the second time, and the central spiral structure still exists, indicating that the cycle is relatively significant. With the development of homology, an increasing number of old homology groups have died, and new homology groups have emerged. In the bottom-left figure, the homology group gradually converges to the three-layer structure, corresponding to the evenly distributed three-layer topology in the bottom-right figure, representing the simple, efficient, and easy-to-use topological features extracted from the original data. In addition, the process fully demonstrates the smoothing and noise reduction functions of the TDA. The node distributions corresponding to each state in Figure 5 are shown in Figure 6.
parameters, \( n_{\text{cubes}} \) is the resolution, representing the number of hypercubes in each dimension. The \( n_{\text{cubes}} \) in chronological order is respectively 30, 15, 10, and 5.

\( n_{\text{nodes}} \) is the number of nodes, representing the number of topology. The \( n_{\text{nodes}} \) in chronological order is respectively 600, 250, 125, and 25.

In this experiment, the Euclidean distance was used as the measure of distance, and other superparameters include \( \text{eps} = 0.1 \), which represents the maximum distance that one of the two samples was considered to be near the other, and \( \text{min}\_\text{samples} = 15 \), indicating the number of samples in the neighborhood where a point is regarded as a core point.

**Figure 5.** Continuous homology process, the time sequence is: top left (1), top right (2), bottom left (3), bottom right (4).

In Figure 6, columns with different colors represent different node types, and the color corresponds to Figure 5. As shown in Figure 6, the change in node type is divided into two stages. In the first stage, with the continuous increase in resolution, the node types increase, and the observed topology becomes increasingly diversified and deeper. For example, at the resolution of 30, a clear spiral structure is observed. In the second stage, with a continuous increase in resolution, the node type decreased, and the dimension of the data decreased. Outliers and noise were gradually fused in the second stage to weaken or even eliminate their influence on the feature representation. For example, when the resolution was 10, the outliers in the lower left corner of Figure 5 were eliminated when the resolution was reduced to five. This also explains the powerful capability of TDA in the field of noise reduction.
Figure 6. Node distribution diagram, the time sequence is: top left (1), top right (2), bottom left (3), bottom right (4).

The result of continuous homology is the birth-death point pair of a series of topological features, and its visual display is shown in Figure 7.

Figure 7. Data persistence diagram (left) and persistence density diagram (right).

In Figure 7, the blue dot represents the birth and death of the 0th order homologous group, and the persistence of the blue dot represents the dispersion of the points in the point cloud. The larger the number of blue dots are near the diagonal, the more dispersed the point cloud. The red dots indicate birth and death in the first-order homologous group. The red dot is far from the diagonal, and the topological property is more significant in the characterization data. Through the analysis of the persistence density map in Figure 7, the order 0 homologous group was frequently born and dies in the interval with a birth time of
(0.015, 0.035), indicating that the data topological characteristics at this stage are unstable, and there are many points on the diagonal on the persistence map.

A persistence diagram can make people intuitively feel the emergence and extinction of topological features and the persistence landscape can be easily combined with mathematical analysis and neural networks.

Any order in a persistent landscape can be used as a topological characteristic. Low-order persistent landscapes contain information on important topological characteristics, whereas high-order persistent landscapes can handle topological noise. Therefore, selecting the order of the persistent landscape as a feature requires maintaining a delicate balance between the loss of important signals and the introduction of excessive noise.

### 5.2. Feature Comparison Experiment

To explore the advantages and disadvantages of the combination of topological features and spatio-temporal features, this study used the GRU model to predict the received traffic value in the next 24 h and uses the original features, PCA features, ICA features, TSNE features, and features extracted by TDA + CNN as training data to carry out the first group of comparative experiments. The experimental results are listed in Table 3. In Table 3, the best and worst values for each evaluation indicators are marked in bold.

| Table 3. Comparison of original features, PCA features, ICA features, TSNE features, TDA + CNN features. |
|---------------------------------------------------------------|
| **MSE** | **R^2** | **MAPE 20% ACC** | **15% ACC** | **10% ACC** |
| Original features | 0.0034 | −0.195 | 0.0845 | 0.9664 | 0.8498 | 0.6998 |
| TSNE characteristics | 0.0031 | −0.0729 | 0.0779 | 0.9832 | 0.8498 | 0.6828 |
| ICA characteristics | 0.0034 | −0.1764 | 0.0839 | 0.9916 | 0.8828 | 0.6078 |
| PCA characteristics | 0.0038 | −0.3069 | 0.0896 | 0.9498 | 0.7996 | 0.6246 |
| TDA + CNN features | 0.0027 | 0.0443 | 0.0707 | 0.9832 | 0.8832 | 0.7162 |

It can be seen from Table 3 that the model trained with TDA + CNN features is significantly better than the model trained with other features in MSE, R^2, 10% ACC, and other evaluation indexes and was similar to the model trained with TSNE features and ICA features in MAPE and 15% ACC but still better than both. The model trained with PCA features achieved the worst score in most indicators, which shows that PCA is not as good as other comparison methods in feature dimensionality reduction of time-series data. It is worth noting that the score of the original feature is also better than that of PCA, which indicates that PCA loses some useful information in the process of dimensionality reduction of the data, resulting in a decline in the prediction performance of the model.

From the perspective of dimension reduction principle, PCA maps high-dimensional data to low-dimensional space through linear projection. TSNE enables similar objects to have a higher probability of being selected. CMPA [27] selects effective features with the help of chaotic sequences. In this paper, spatial features are combined with temporal features, which obviously have stronger abilities in feature expression.

In conclusion, the TDA + CNN feature training model achieved the best score on most evaluation indexes, which proves the feasibility of applying this method to time-series prediction and the powerful ability of this method in feature extraction and data dimensionality reduction.

### 5.3. Model Comparison Experiment

To explore the advantages and disadvantages of the TCAG model, this study used the features extracted by TDA + CNN to train MLP, TCN, LSTM, GRU, TCAG, and other neural network models for comparative experiments.

In Table 4, the best and worst values for each evaluation indicators are marked in bold. Amongst the four models compared, the R^2 value of the MLP model was the smallest, indicating that there were complex changes in the data, resulting in poor fitting of the model. The R^2 score of TCAG is the only value greater than 0, which further explains...
the complexity of the data and shows that the TCAG model can achieve a better fitting effect than other benchmark models. The average absolute percentage error of TCAG was 7.56%, 8.96% higher than that of the worst MLP. Compared with other benchmark models, they satisfy higher accuracy requirements. As shown in Table 4, the performances of TCN, LSTM, and GRU are similar in various indices, which are located between MLP and TCAG. In addition, it can be seen from Figure 8 that the accuracy of TCAG does not decline with the increase of prediction steps, which indicates that the model has strong generalization ability and can learn the historical laws of data well.

Table 4. Comparison of MLP, TCN, LSTM, GRU, TCAG.

|       | MSE  | R²   | MAPE | 20% ACC | 15% ACC | 10% ACC |
|-------|------|------|------|---------|---------|---------|
| MLP   | 0.0344 | -1.1096 | 0.1652 | 0.699   | 0.499   | 0.283   |
| TCN   | 0.0066 | -1.2635 | 0.1065 | 0.883   | 0.766   | 0.574   |
| LSTM  | 0.0044 | -0.5298 | 0.0916 | 0.916   | 0.783   | 0.561   |
| GRU   | 0.0057 | -0.9725 | 0.1105 | 0.824   | 0.691   | 0.533   |
| TCAG  | 0.003  | 0.0401 | 0.0756 | 0.958   | 0.874   | 0.724   |

Figure 8. TCAG model prediction results.

It can be seen from Figures 8–12 that TCAG has the best fitting effect among the five models, and only two of the predicted 24 points exceeded the range of 15% ACC. The MLP model has the worst fitting effect, with a large number of predicted values exceeding the range of 15% ACC, and the error increases with an increase in the prediction step size. The TCN model misestimates the fluctuation range of the value in the range of the step size [4,9] but then returns to a more normal state. The effect of LSTM is close to that of the GRU and second only to the TCAG model.
Finally, the dynamic threshold interval is calculated according to Formula (2), as shown in Figure 13. The blue solid line is the actual value, the red solid line is the predicted value, and the blue shaded area represents the dynamic threshold interval at 95% confidence. The upper and lower boundaries of the area are used as the threshold to trigger the alarm. When the real value exceeds the threshold, it can be determined that there is a 95% probability that the equipment is abnormal, and the alarm should be sent immediately.

Figure 8. TCAG model prediction results.

Figure 9. MLP model prediction results.

Figure 10. TCN model prediction results.

Figure 11. LSTM model prediction results.

Figure 12. GRU model prediction results.
Figure 12. GRU model prediction results.

In summary, the TCAG model achieved the best score for all evaluation indicators, which proves that the model is superior to the four benchmark models in terms of prediction accuracy and robustness and proves the correctness of selecting the GRU model as the TCAG model component.

Finally, the dynamic threshold interval is calculated according to Formula (2), as shown in Figure 13. The blue solid line is the actual value, the red solid line is the predicted value, and the blue shaded area represents the dynamic threshold interval at 95% confidence. The upper and lower boundaries of the area are used as the threshold to trigger the alarm. When the real value exceeds the threshold, it can be determined that there is a 95% probability that the equipment is abnormal, and the alarm should be sent immediately.

Figure 13. Dynamic threshold interval diagram.

6. Conclusions and Prospect

This paper proposes a TCAG model that uses a convolutional neural network to extract the spatio-temporal features, uses topological data analysis to extract the topological features, and then combines the spatio-temporal features and topological features to train the attention GRU neural network model. Finally, according to the prediction results of the GRU model, the dynamic threshold interval was calculated for the early fault warning of
the information equipment. Based on the experimental results, the following conclusions can be drawn:

(1) The proposed dynamic threshold-setting mechanism solves the problem of early fault warning in AIOps. The mechanism calculates the threshold interval automatically adjusted with scene change according to the prediction results and can fully consider the prediction error caused by step growth, which has great practical significance.

(2) Traditional feature extraction methods are difficult to adapt to instability and nonlinear structures in a time series. To solve this challenge, spatiotemporal features and topological features were combined to train the attention GRU neural network model. Compared with the original, TDA, ICA, and TSNE features, the TCAG model was able to extract the most concise and effective feature expression in the data and significantly improve the prediction effect of the model.

(3) To verify the prediction performance of the TCAG model, comparative experiments were carried out using MLP, TCN, LSTM, and GRU models. The results showed that the TCAG model is superior to the benchmark model in various evaluation indexes and obtains the best prediction accuracy and robust performance. It is worth mentioning that the average absolute percentage error increased by 8.96% compared with that of the MLP model.

In this study, the early fault warning of AIOps was extensively studied and achieved ideal results. Future work will explore other directions of AIOps (abnormal monitoring, fault self-healing, etc.) and will continue to contribute to the improvement of the intelligent degree of operation and maintenance.

Author Contributions: Methodology, Z.C.; Supervision, L.Z.; Writing—original draft, K.W.; Writing—review & editing, Y.T. and J.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by The Major Program of the National Natural Science Foundation of China (71633006); Technological Innovation 2030–Subproject of “New Generation Artificial Intelligence” Major Project (2020AAA0109605).

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
1. Everything You Need to Know About AIOps. Available online: https://www.moogsoft.com/resources/aiops/guide/everything-aiops/ (accessed on 12 February 2019).
2. Li, L.; Hansman, R.J.; Palacios, R.; Welsch, R. Anomaly Detection via a Gaussian Mixture Model for Flight Operation and Safety Monitoring. Transp. Res. Part C Emerg. Technol. 2016, 64, 45–57. [CrossRef]
3. Farshchi, M.; Schneider, J.; Weber, I.; Grundy, J. Metric Selection and Anomaly Detection for Cloud Operations Using Log and Metric Correlation Analysis. J. Syst. Softw. 2017, 137, 531–549. [CrossRef]
4. Nedelkoski, S.; Cardoso, J.; Kao, O. Anomaly Detection from System Tracing Data Using Multimodal Deep Learning. In Proceedings of the 2019 IEEE 12th International Conference on Cloud Computing (CLOUD), Milan, Italy, 8–13 July 2019; pp. 179–186.
5. Wang, W.; Lyu, G.; Shi, Y.; Liang, X. Time Series Clustering Based on Dynamic Time Warping. In Proceedings of the 2018 IEEE 9th International Conference on Software Engineering and Service Science (ICSESS), Beijing, China, 23–25 November 2018; pp. 487–490. [CrossRef]
6. Wen, L.; Gao, L.; Li, X. A New Deep Transfer Learning Based on Sparse Auto-Encoder for Fault Diagnosis. IEEE Trans. Syst. Man Cybern. Syst. 2019, 49, 136–144. [CrossRef]
7. Wang, Z.; Zhang, M.; Wang, D.; Song, C.; Liu, M.; Li, J.; Lou, L.; Liu, Z. Failure Prediction Using Machine Learning and Time Series in Optical Network. Opt. Express 2017, 25, 18553–18565. [CrossRef] [PubMed]
8. Cao, H.; Long, F.; Wang, B.; Peng, X.; Chen, X.; Tao, X. DeepFaultNet: Predicting fault severity of communication cable with Hybrid-ResCNN. IEEE Trans. Circuits Syst. II Express Briefs 2022. [CrossRef]
9. Chen, Q.; Zheng, Z.; Hu, C.; Wang, D.; Liu, F. Data-Driven Task Allocation for Multi-Task Transfer Learning on the Edge. In Proceedings of the 2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS), Dallas, TX, USA, 7–10 July 2019; pp. 1040–1050.
10. Jin, M.; Lv, A.; Zhou, Y.; Wen, Z.; Zhong, Y.; Zhao, Z.; Wu, J.; Li, H.; He, H.; Chen, F.; et al. An Anomaly Detection Algorithm for Microservice Model Based on Robust Principal Component Analysis. *IEEE Access* 2020, 8, 226397–226408. [CrossRef]
11. Soualhi, A.; Medjaher, K.; Zerhouni, N. Bearing Health Monitoring Based on Hilbert–Huang Transform, Support Vector Machine, and Regression. *IEEE Trans. Instrum. Meas.* 2015, 64, 52–62. [CrossRef]
12. Zhao, N.; Zhu, J.; Wang, Y.; Ma, M.; Zhang, W.; Liu, D.; Zhang, M.; Pei, D. Automatic and Generic Periodicity Adaptation for KPI Anomaly Detection. *IEEE Trans. Netw. Serv. Manag.* 2019, 16, 1170–1183. [CrossRef]
13. Wang, H.; Zhang, H. AIOPS Prediction for Hard Drive Failures Based on Stacking Ensemble Model. In Proceedings of the 2020 10th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, NV, USA, 6–8 January 2020; pp. 417–423. [CrossRef]
14. Khalil, K.; Eldash, O.; Kumar, A.; Bayoumi, M. Machine Learning-Based Approach for Hardware Faults Prediction. *IEEE Trans. Circuits Syst. I Regul. Pap.* 2020, 67, 3880–3892. [CrossRef]
15. Wen, L.; Li, X.; Gao, L.; Zhang, Y. A New Convolutional Neural Network-Based Data-Driven Fault Diagnosis Method. *IEEE Trans. Ind. Electron.* 2018, 65, 5990–5998. [CrossRef]
16. Gensler, A.; Henze, J.; Sick, B.; Raabe, N. Deep Learning for solar power forecasting—An approach using AutoEncoder and LSTM Neural Networks. In Proceedings of the 2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Budapest, Hungary, 9–12 October 2016; pp. 2858–2865. [CrossRef]
17. Kong, W.; Dong, Z.Y.; Jia, Y.; Hill, D.J.; Xu, Y.; Zhang, Y. Short-Term Residential Load Forecasting Based on LSTM Recurrent Neural Network. *IEEE Trans. Smart Grid* 2019, 10, 841–851. [CrossRef]
18. Chen, K.; Chen, K.; Wang, Q.; He, Z.; Hu, J.; He, J. Short-Term Load Forecasting with Deep Residual Networks. *IEEE Trans. Smart Grid* 2019, 10, 3943–3952. [CrossRef]
19. Baolong, L.; Zhe, L. High dimensional data hiding pattern mining based on topological data analysis. *J. Syst. Simul.* 2019, 31, 1755–1762.
20. Ravishanker, N.; Renjie, C. Topological Data Analysis (TDA) for Time Series. *arXiv* 2019, arXiv:1909.10604.
21. Khasawneh, F.A.; Munch, E. Chatter detection in turning using persistent homology. *Mech. Syst. Signal Process.* 2016, 70, 527–541. [CrossRef]
22. Umeda, Y. Time series classification via topological data analysis. *Inf. Media Technol.* 2017, 12, 228–239. [CrossRef]
23. Seversky, L.M.; Davis, S.; Berger, M. On time-series topological data analysis: New data and opportunities. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 59–67.
24. Yang, A.; Zhuansun, Y.; Shi, Y.; Liu, H.; Chen, Y.; Li, R. IoT System for Pellet Proportioning Based on BAS Intelligent Recommendation Model. *Trans. Ind. Inform.* 2021, 17, 934–942. [CrossRef]
25. Ren, Y.; Teng, C.; Li, F.; Chen, B.; Ji, D. Relation classification via sequence features and bi-directional LSTMs. *Wuhan Univ. J. Nat. Sci.* 2017, 22, 489–497. [CrossRef]
26. Yang, A.; Zhang, C.; Chen, Y.; Zhuansun, Y.; Liu, H. Security and Privacy of Smart Home Systems Based on the Internet of Things and Stereo Matching Algorithms. *IEEE Internet Things J.* 2020, 7, 2521–2530. [CrossRef]
27. Alrasheedi, A.F.; Alnowibet, K.A.; Saxena, A.; Sallam, K.M.; Mohamed, A.W. Chaos Embed Marine Predator (CMPA) Algorithm for Feature Selection. *Mathematics* 2022, 10, 1411. [CrossRef]