

Article

Dynamic Anomaly Detection in Gantry Transactions Using Graph Convolutional Network-Gate Recurrent Unit with Adaptive Attention

Fumin Zou ¹, Yue Xing ¹, Qiang Ren ², Feng Guo ^{1,*}, Zhaoyi Zhou ¹ and Zihan Ye ¹

- ¹ Fujian Key Laboratory for Automotive Electronics and Electric Drive, Fujian University of Technology, Fuzhou 350118, China; fmzou@fjut.edu.cn (F.Z.); 2211801014@smail.fjut.edu.cn (Y.X.); 2211301019@smail.fjut.edu.cn (Z.Z.); 2231901003@smail.fjut.edu.cn (Z.Y.)
- ² Chongqing Key Laboratory of Public Big Data Security Technology, Chongqing College of Mobile Communication, Chongqing 401420, China; rqi202204@163.com
- * Correspondence: mapli@fjut.edu.cn; Tel.: +86-157-0599-1258

Abstract: With the wide application of Electronic Toll Collection (ETC) systems, the effectiveness of the operation and maintenance of gantry equipment still need to be improved. This paper proposes a dynamic anomaly detection method for gantry transactions, utilizing the contextual attention mechanism and Graph Convolutional Network-Gate Recurrent Unit (GCN-GRU) dynamic anomaly detection method for gantry transactions. In this paper, four different classes of gantry anomalies are defined and modeled, representing gantries as nodes and the connectivity between gantries as edges. First, the spatial distribution of highway ETC gantries is modeled using the GCN model to extract gantry node features. Then, the contextual attention mechanism is utilized to capture the recent patterns of the dynamic transaction graph of the gantries, and the GRU model is used to extract the time-series characteristics of the gantry nodes to dynamically update the gantry leakage. Our model is evaluated on several experimental datasets and compared with other commonly used anomaly detection methods. The experimental results show that our model outperforms other anomaly detection models in terms of accuracy, precision, and other evaluation values of 99%, proving its effectiveness and robustness. This model has a wide application potential in real gantry detection and management.

Keywords: ETC gantry; dynamic anomaly detection; GCN-GRU; attention mechanism; graph data



Citation: Zou, F.; Xing, Y.; Ren, Q.; Guo, F.; Zhou, Z.; Ye, Z. Dynamic Anomaly Detection in Gantry Transactions Using Graph Convolutional Network-Gate Recurrent Unit with Adaptive Attention. *Appl. Sci.* **2023**, *13*, 11068. <https://doi.org/10.3390/app131911068>

Academic Editors: Long Wen, Zhuyun Chen, Haisong Huang and Jianan Wei

Received: 21 September 2023
Revised: 4 October 2023
Accepted: 6 October 2023
Published: 8 October 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/>).

1. Introduction

With the continuous development of intelligent transportation technology, the Electronic Toll Collection (ETC) gantry system has been widely used on highways [1,2]. As of the end of 2022, a total of 28,328 physical gantries and 37,947 dedicated ETC lanes have been established in China, with nearly one billion ETC transactions recorded daily [3]. However, since the system involves a large number of devices and data transmission, the occurrence of system failures may lead to problems such as traffic congestion and loss of fees. Therefore, the efficient dynamic anomaly detection of ETC gantry can help maintenance personnel to quickly detect and solve system anomalies, ensuring the normal operation of the ETC gantry system, which is of great significance, and can also provide a data basis and guarantee for a series of research [4–7] on expanding the application of smart highways based on ETC transaction data. The original ETC gantry anomaly detection mainly uses manual inspection or remote detection. However, these methods have some problems: the human resources' investment is large, and traditional abnormality detection mainly relies on the experience of personnel, making it prone to omission or misdiagnosis, and it is difficult to meet the requirements of real-time operation and accuracy. Highway companies in various provinces and cities in China have also carried out a series of studies

on the intelligent operation and maintenance of ETC gantries. Zheng et al. [8] improved and optimized the modules of the system control cabinet. This provides a safe and stable operating environment for the ETC gantry system in terms of front-end sensing, storage, power supply, and communication. Jin et al. [9] briefly described intelligent detection and operation and maintenance, focusing on two aspects, big data fusion analysis and detection, and operation and maintenance full life-cycle management, in order to improve the detection, operation, and maintenance business capacity of highways. Wei et al. [10] developed a toll system key power, network cloud detection, and warning system; the system can raise alarms for abnormalities caused by the interruption of the gantry cable or the tripping of any switch.

The operation and maintenance of the gantry system presented above are based on the indicators of the hardware equipment, which are to detect faults and then realize the operation and maintenance management of the gantry. However, this method is rather one-sided and only analyzes the local anomalies in the system power and network. Moreover, most use experience and theory to design the operation and maintenance system, which have poor reliability and cannot effectively meet the needs of gantry operation and maintenance in a timely manner. Therefore, this paper summarizes the research work in the field of equipment anomaly detection in recent years and, on this basis, combines the characteristics of highway ETC transaction data, and proposes a highway ETC gantry anomaly dynamic detection method based on the combination of spatio-temporal characteristics. The main contributions are specified as follows:

- The paper proposes, for the first time, the use of gantry transaction data for gantry anomaly detection, which provides a comprehensive assessment of gantry anomalies at the macro-level and defines four categories of gantry anomalies of different grades, and this categorization system allows for the continuous observation of potentially anomalous risky gantries.
- The method uses a combination of a Graph Convolutional Network (GCN), contextual attention mechanism, and Gate Recurrent Unit (GRU) to not only consider the spatial distribution of gantries, but also capture the temporal dynamic properties of gantry transaction data. The time-series characteristics of the gantries are fully considered, leading to better detection and updating of the gantries' status.
- This paper carries out a large number of experiments using real ETC data, and achieves more than 99% accuracy in all kinds of evaluation indexes, showing a better performance than other baseline models, and the model shows good performance with different data volumes, which proves the effectiveness of this paper's method.

The rest of the paper is organized as follows: Section 2 provides an overview and summary of related work in the field of anomaly detection. Section 3 presents a problem description and related definitions for the dynamic detection of gantry anomalies. Section 4 presents the proposed model and methodology. Section 5 conducts detailed experiments on the developed model and presents the experimental results and the performance of the proposed model. Finally, Section 6 summarizes the work of this paper and provides an outlook for future work.

2. Literature Review

ETC gantry is a system of multiple hardware devices integration, which mainly contains front-end devices, such as industrial controllers, power adapters, RSU antennas, license plate recognition devices, and back-end devices, such as servers and switches. Current research on hardware device anomaly detection is broadly categorized into three main groups: signal processing methods, multivariate statistics-based methods, and machine-learning-based methods.

Among the methods of signal processing, time-frequency analysis and sparse filtering can improve the performance of fault diagnosis models [11–14]. Li [15] analyzed the spectrum of the signal by applying dimensionality reduction to the spectral image obtained by fast Fourier transform for fault diagnosis. The proposed wavelet transform overcomes

the problem of localized characteristics and is used in fault diagnosis at a non-smooth multiscale, but cannot further decompose signals in high-frequency bands. Among the multivariate statistical analysis methods, Principal Component Analysis (PCA) has an excellent advantage for data dimensionality reduction processing. Liang et al. [16] proposed an anomaly detection method based on generalized mutual entropy principal element analysis and applied it to the Tennyson–Eastman process for anomaly detection; the method shows good performance in handling non-Gaussian anomaly detection, with low false-alarm and miss rates. Machine learning methods have received more and more attention in anomaly detection and diagnosis techniques due to their good self-learning, recognition, and classification capabilities. Shao et al. [17] proposed an AM-Relief feature selection using an integration (support vector machine (SVM)) diagnostic algorithm; the proposed method increases the fault diagnosis accuracy from 83.0% to 98.9%, providing a new idea for research on the mechanical fault diagnosis of high-voltage circuit breakers. Yang et al. [18] proposed a bearing fault diagnosis model based on multi-sensor information fusion for aero-engine bearings; the accuracy of this method is improved by 36.92% compared to the classification and identification of faults using SVM. Liu et al. [19] proposed a train speed anomaly detection method that combines extreme gradient boosting and anomaly verification methods; the method has high performance and meets the requirements of the real-time detection of train operations. However, the use of the XGboost model in the research methodology may have some limitations, such as a strong dependence on parameter settings, which may require more tuning and validation. The equipment anomaly detection methods mentioned above are mainly applied to industrial processes, machinery and equipment. They have not been fully applied in the anomaly detection of ETC gantry systems and are limited by the limitations of each system device, which can only be detected individually through the design of specific algorithms for each device.

The transaction data generated by the ETC gantry system contain time-series properties, and the topology consisting of vehicle trajectories extracted from the transaction data has graph network structure properties. Gantry anomalies can cause missing transaction data, which manifest as point anomalies in time series anomalies and structural anomalies in topological road networks, so gantry anomalies can be investigated using dynamic graph anomaly detection methods based on time series and the structural properties of road network structural characteristics graph network structure characteristics. Dynamic graph anomaly detection is mainly categorized into isolated individual anomaly detection, anomaly group detection and event anomaly detection.

The main types of dynamic graph-oriented anomalous individual detection are node anomalies, anomalous edges, and anomalous moments. Ranshous et al. [20] proposed a multidimensional count min sketch (CM-Sketch) approach to maintain the count of edges, which mainly focuses on the detection of anomalies on the edge flow, modeling the dynamic network as edges arriving over time and detecting the anomalous edges on this basis. However, empirical metrics are used, making the method less able to generalize. Eswaran et al. [21] considered edges connecting sparsely connected regions occurring in emergency situations as anomalous edges and compared the connectivity between nodes around the edges before and after the addition of edges, such that edges with increased connectivity have higher anomaly values. However, this method requires extra space to maintain the sample and does not detect exceptions in the general case. Dynamic graph anomaly population detection requires consideration of the time dimension, and different areas of concern for the type of anomalous groups are different, such as those that can be analyzed in traffic or used to analyze anomalous groups in a social network. Zhang et al. [22] proposed the FGN detection method based on graph neural network to resolve the problem of abnormal user detection in social networks. The article proposes HNR, a graph node representation learning method incorporating hyperbolic geometry, which reduces the training complexity and improves the information transfer process, preserving higher-order neighbor information. However, the HNR in the article only utilizes the node's own attributes and does not make full use of the neighbors' attribute information, meaning that

it has a limited characterization learning ability. Event anomaly detection is a method of anomaly detection that looks for points in time series data that are different from most other points in time, and then delves deeper by using static graphs of network structure data for the points in time. Neural network anomalous event detection is widely used, especially the graph neural network, which is developed for deep learning on data with a graph structure, which play an important role in many directions [23–25]. Liu et al. [26] proposed the TADDY framework to represent a dynamic graph as a series of discrete graph snapshots, and defined the anomaly detection problem as one that provides an anomaly score for each edge at each timestamp. However, the overall training efficiency of the method is low, and it is not friendly to large-scale dynamic graphs. Ding et al. [27] proposed AnoGLA, a network anomaly detection framework based on graph neural networks and long- and short-term memory networks. The framework views network traffic as a subgraph, extracts graph structural features using GCN, learns traffic variation patterns over time using long- and short-term memory networks, and finally generates anomaly scores for each traffic flow through end-to-end training using a negative sampling training framework. However, the paper only considered a small number of attributes for node coding, did not fully explore the node information, and was not effective for attribute-rich datasets.

In summary, the current research related to the abnormal detection of ETC gantry equipment is still relatively shallow; most of the research is more focused on engineering technology, and there is still a lot of room for academic research. Graph-based dynamic anomaly detection has shown great advantages in the field of big data, is capable of dealing with heterogeneous structures and time dependencies in dynamic networks, and is now widely used in such fields as finance, Internet security, social relationship mining, and telemarketing fraud detection. However, no scholars have applied the method to the dynamic detection of gantry equipment to date. Therefore, to address the shortcomings of the current research related to the abnormal dynamic detection of gantries, as well as the characteristics of the gantry transaction data and the properties of the graph network model, this paper proposes the GCN-GRU gantry anomaly dynamic detection method based on the contextual attention mechanism.

3. Relevant Definitions and Problem Descriptions

Definition 1. (*highway segment QD*): Two adjacent gantries on a highway form a QD, with gantry $Node_1$ as the start node of the QD and $Node_2$ as the end node:

$$QD = \langle Node_1, Node_2 \rangle. \quad (1)$$

Definition 2. (*highway section LD*): Highway segments consist of multiple adjacent QDs:

$$LD = \{QD_1, \dots, QD_n\} \quad (2)$$

where the starting node of QD_1 is called the starting point of the section, the terminating node of QD_n is called the end point of the section, and the end point of the previous QD is the starting point of the next QD.

Definition 3. (*Freeway Road Network LW*): All segments of the LD within the freeway study area comprise the freeway road network:

$$LW = \{LD_1, \dots, LD_n\}. \quad (3)$$

Definition 4. *Effective transaction topology*: In Figure 1, Figure 1a shows the local LW, and Figure 1b–d are the three sets of locally effective transaction topologies. In Figure 1b there are two ordered trading sections, $\langle K, H, I \rangle$ and $\langle K, H, G, F, E \rangle$. As shown in Figure 1e, if the same license

plate number appears in the transaction data of both gantry G and gantry E in the time series, but there is no same license plate number in gantry F, this means that gantry F has a transaction failure.

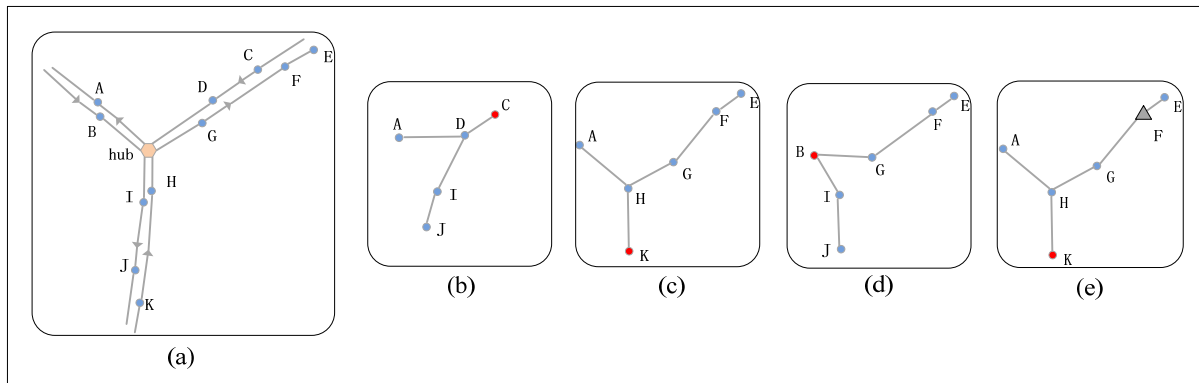


Figure 1. Effective transaction topology diagram. (a) Schematic diagram of local road network, Where the yellow dots represent hubs, the blue dots represent gantry nodes, and the letters A, . . . , K, denote the names of the portal frame gantry nodes. (b) Effective transaction topology diagram, where blue dots indicate gantry nodes and red dots indicate the starting point of the effective topology. (c) Effective transaction topology diagram, (d) Effective transaction topology diagram, (e) Schematic representation of transaction failures in an efficient transaction topology, where the gray triangles indicate the gantry nodes where the transaction failed.

Definition 5. *Gantry Anomaly:* A pattern of trading in which the gantry exhibits significantly different trading behavior from normal trading behavior during a specific time window. Based on the continuity of anomaly occurrence, gantry anomalies can be subdivided into the following four categories.

1. Burst Anomaly (BA): refers to multiple discrete transaction failures for different vehicle transactions on the gantry within a time window. Defined as more than B discrete transaction failures on the gantry within a time window T , where B is a predetermined threshold.
2. Mild Continuous Anomaly (MCA): refers to a time window in which K_1 consecutive different vehicles on the gantry have failed to trade, where K_1 is a predetermined threshold.
3. Moderate Continuous Anomaly (MoCA): refers to a time window in which there are K_2 consecutive different vehicles on the gantry that have failed to trade, where $K_2 > K_1$.
4. Severe Continuous Anomaly (SCA): refers to a time window in which K_3 or more consecutively different vehicles on the gantry have failed to trade, where $K_3 > K_2$.

In this paper, we distinguish the severity of anomalies by defining the thresholds B , K_1 , K_2 , and K_3 for the number of failed transactions, and capture the temporary or persistent nature of the anomaly based on the time window T .

Problem description:

Combining the topology of the highway network, the gantry nodes are taken as the nodes of the graph, the gantry transactions are taken as the edges of the graph, the dynamic detection of the gantry anomalies is converted to the dynamic anomaly detection of the graph nodes, and based on the four types of anomalies defined above, the problem is finally transformed into the dynamic anomaly detection based on the node classification task. The set of n time slices $T = \{t_1, t_2, t_3, \dots, t_n\}$ is divided according to the temporal characteristics of the transaction data and the zone travel time, the set of edges E of the subgraph is the transactions between the gantries, E is the set of edges of the subgraph, and the adjacency matrix of the graph G is denoted as $A = (A_{t_1}, A_{t_2}, \dots, A_{t_n})$, $A \in R^{N \times N}$. Using the information of the transaction data as attribute features of the nodes, the feature

matrix of the graph stream is denoted as $X = (X_{t_1}, X_{t_2}, \dots, X_{t_n})$, $X_{t_i} \in R^{N \times P}$, where P is the dimension of the attribute characterizing the node. Then, the feature graph at time step t is denoted as $G_t = (X_t, A)$; the embedding of the node at time step t in G_t is denoted as $Y_{v_t} = f_{embed}(X_{v_t})$. The anomaly scoring function f_{score} is used to compute the anomaly scores S of the node at different time steps, and finally, based on the anomaly scores $score_v$ and the threshold c , whether the node v contains the anomaly in Definition 5 is determined.

4. Methodology

The gantry anomaly dynamic detection algorithm proposed in this paper mainly includes three parts: graph convolutional neural network, attention mechanism, and GRU, and the overall framework is shown in Figure 2. The algorithm first uses the GCN model to model the transaction data and adjacency of the gantry, learns the connection relationship between the gantry nodes and the influence of the transaction data through the GCN, and fuses the node features with the graph structure to generate a new node representation. Then, time-series modeling is performed with the GRU model to capture the time-dependence of the same gantry transactions, which allows for dynamic adjustment of the hiding state and the detection of various anomalies through the computation of reset gates and update gates. Finally, the classification results are obtained through the fully connected layer combined with the SoftMax function.

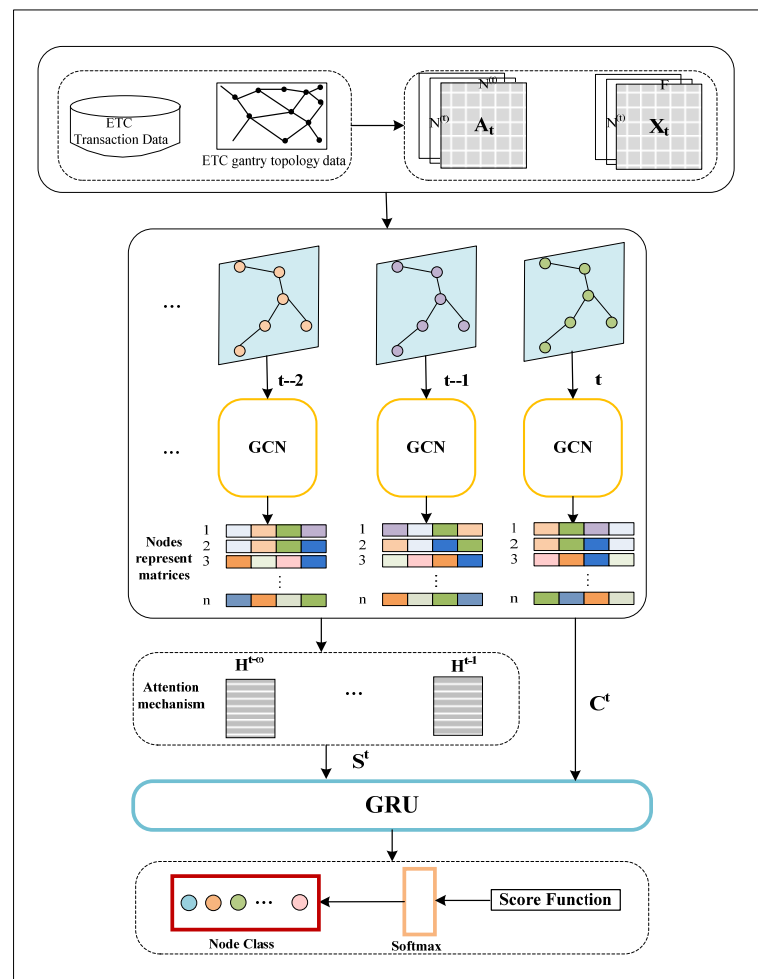


Figure 2. Overall framework, where different colors indicate different gantry nodes.

4.1. Graph Convolutional Neural Network Module

The main focus of this study is on gantry anomalies, which are primarily characterized by missed transactions. As defined in Section 3, it is evident that the determination of

gantry missed transactions is influenced by the transaction history of neighboring gantry nodes. Furthermore, ETC gantries on highways exhibit varying topological relationships in different sections. It is clear that the mutual influence between ETC gantries with different topological relationships will vary accordingly. If we can effectively extract and utilize the topological relationships between ETC gantries, the detection of gantry anomalies will be more accurate.

Conventional Convolutional Neural Networks (CNNs) can only handle spatial data with regular Euclidean structures and are incapable of processing irregular non-Euclidean spatial data [28]. Therefore, the GCN model was proposed to handle non-Euclidean spatial data. The spatial distribution of ETC gantries on highways constitutes a non-Euclidean geometric structure. Hence, we employ the GCN model to model the spatial distribution of ETC gantries on highways and extract gantry node features.

To begin with, we construct a transaction graph for each time interval using the gantry transaction data, where gantries are considered as nodes, and if there is a transaction between the same vehicle and two gantries, then there is an edge between those two gantries. Assuming we have a transaction dataset $Tr = (v_i, c_i, t_i)$, where v_i represents the gantry number, c_i represents the license plate number, and t_i represents the transaction time. The gantry transaction graph is represented as a graph $G = (V, E)$, where there are n gantry nodes, and m transaction edges between gantries. Here, $V = \{v_1, v_2, \dots, v_n\}$ represents the set of gantry nodes, and $E = \{(v_i, v_j, w_k)\}$ represents the set of edges extracted from the transaction dataset D , where v_i and v_j denote adjacent gantry numbers. The gantry transaction network using an adjacency matrix:

$$A_{i,j} = \begin{cases} 1 & \text{if there is an edge between node } i \text{ and node } j \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

where $A_{i,j}$ indicates whether there is a transaction edge between gantry nodes i and j . If a transaction edge exists, then $A_{i,j} = 1$; otherwise, $A_{i,j} = 0$. Construct the gantry transaction graph according to Algorithm 1:

Algorithm 1: Time Slice Transaction Graph Construction Method

Input: Gantry topology dataset, vehicle transaction dataset;

Output: Graph data

1. # Define the number of gantry nodes and the feature vector dimension
 2. num_nodes = 6, feat_dim = 32
 3. node_list, edge_list = extract_node_and_edge_info(RSU_data, OBU_data)
 4. # Construct the gantry node feature vectors
 5. node_feats = {}
 6. for i in range(num_nodes):
 7. node_feats[i] = np.zeros(feat_dim)
 8. # Define the adjacency matrix of a graph
 9. adj_mat = np.zeros((num_nodes, num_nodes))
 10. # Read gantry transaction data
 11. transactions = read_transactions()
 12. # Iterate over all transaction data and update the adjacency matrix and node feature vectors
 13. for trans in transactions:
 14. node_id = trans['node_id'] # Get node numbers
 15. node_feats[node_id][trans['feat_index']] = trans['feat_value'] # update the node feature vectors
 16. for adj_node_id in trans['adj_nodes']: # Iterate over neighboring nodes
 17. adj_mat[node_id][adj_node_id] = 1 # Updating the adjacency matrix
 18. # Building a gantry trade chart
 19. graph = Graph(adj_mat, node_feats)
-

In the above gantry transaction graph construction process, this paper uses the normalization method [29] to process the adjacency matrix so that it maintains symmetry in information transfer. The symmetric normalization of the adjacency matrix is performed as follows:

$$\hat{A}^t = \hat{D}^{-\frac{1}{2}} \tilde{A}^t \hat{D}^{-\frac{1}{2}}, \tilde{A}^t = A^t + I_N, \tilde{D}_{i,i} = \sum_{j=1}^n \tilde{A}_{i,j}^t \tag{5}$$

where \hat{A}^t is the symmetrically normalized adjacency matrix, and during the process of capturing adjacent node features, $\tilde{A}^t = A^t + I_N$ prevents the current node’s feature information from being unable to propagate. D is the degree matrix, where $D_{i,i}$ represents the degree of gantry node i , i.e., the number of edges connected to it.

After generating the adjacency matrix $A = \{A_{t_1}, A_{t_2}, A_{t_3}, \dots, A_{t_n}\}$ as well as the feature matrix $X = (X_{t_1}, X_{t_2}, \dots, X_{t_n})$, $X_{t_i} \in R^{N \times P}$ of the time-slice transaction graph, we learn the representation of the graph using GCN. For each time slice, the neighbor information of the node is aggregated in the output vector. The GCN can be trained by a back propagation algorithm and the relationships between node features are learned in the network.

Taking a time-slice transaction graph as an example, the GCN computes the embedding vectors of each node by performing a weighted average of the features of the neighboring nodes. This process can be viewed as a series of convolution operations. For each node i , its embedding vector can be expressed as:

$$h_i^l = \sigma \left(\sum_{j \in N(i)} \frac{1}{c_{i,j}} h_j^{l-1} W^{(l-1)} \right) \tag{6}$$

where $N(i)$ is the set of neighboring nodes of node i , c_{ij} is the normalization factors of the edges between nodes i and j , $W^{(l-1)}$ is the parameter matrix from layer $l - 1$ to layer l , $\sigma(\cdot)$ is the activation function ReLU.

As shown in Figure 3, each node in the figure represents the highway ETC gantry, which demonstrates that the GCN model captures the features of the current node and its neighboring nodes through a convolutional layer, and ultimately outputs the current node’s state matrix $C^t = GCN_L(Z^l)$ that aggregates information of the neighboring nodes, GCN_L represents the L -layer GCN, and the specific convolutional process of GCN_L is as follows:

$$Z^l = ReLU(\hat{A}^t Z^{(l-1)} W^{(l-1)}) \tag{7}$$

$$C^t = ReLU(\hat{A}^t Z^{(l-1)} W^{(l-1)}) \tag{8}$$

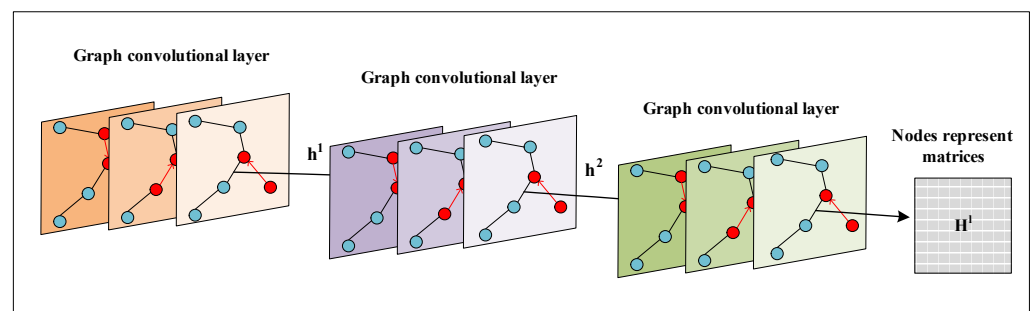


Figure 3. Schematic diagram for feature aggregation of node transaction graphs within time slices, where the red dots indicate the current node and the neighboring nodes to be aggregated by the current convolutional layer, and the red arrows indicate the neighboring nodes to be aggregated towards the current node.

4.2. Attention Mechanism

After obtaining the information of the aggregated neighbor nodes of each time transaction graph, in order to capture the dynamic information of the transaction graphs of different time slices, in this paper, we propose a localized proximity time window state, construction based on a contextual attention mechanism, as shown in Figure 4. In this attention mechanism, each time step t is represented as three aspects:

1. The query vector q_t is used to represent the focus or information of interest at the current time step t . It captures the feature representation of the current time step and is used as a query for similarity calculation with the keys of other time steps.
2. The key vector k_t is used to represent the information of other time steps. It contains the target for similarity computation with the query vector q_t for the current time step t . The key vector can be regarded as a representation of other time steps for comparison with the query of the current time step.
3. The value vector v_t contains the feature representation or information for each time step. It comprises a sequence of vectors used to compute the attention weights. The value vector can contain important information related to the transaction graph.

At each time step t , the similarity between the query vector q_t and the key vector k_t is usually computed using the dot product, and additive attention (using fully connected layers) method. After the similarity is calculated, it is normalized by softmax function to obtain the attention weight α_t ; these weights indicate the most relevant position to the query q_t at the current position t . Weighted summation of a sequence of value vectors v_t using attention weights yields the context vector c_t . c_t contains information about the query vector q_t at the current time step t , which is used as a feature representation of the current time step to capture information about the dynamics of the transaction graph during a locally proximate time window.

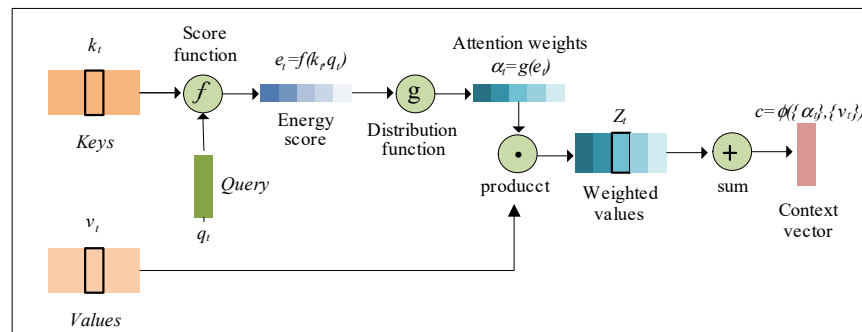


Figure 4. Schematic framework of the attention mechanism.

Based on the above framework principle of the attention mechanism, Q_h corresponds to Query in the graph, which is used to map the approaching time window state to the feature of attention at the current location. $C_{h,i}^t$ corresponds to Key in the graph denoting the approaching time window state of the current node i at time step t , which is used to compute the relevance degree. The result Value processed by the mapping function represents the approaching time window state of the current node i at time step t . The feature representation obtained after processing through the mapping function Q_h is used to generate the context representation. The specific calculation process is as follows:

$$C_{h,i}^t = [h_i^{t-\omega}; \dots; h_i^{t-1}] \quad C_{h,i}^t \in R^{\omega \times n} \quad (9)$$

$$e_{h,i}^t = r^T \tanh(Q_h(C_{h,i}^t)^T) \quad e_{h,i}^t \in R^{\omega} \quad (10)$$

$$a_{h,i}^t = \text{softmax}(e_{h,i}^t) \quad a_{h,i}^t \in R^{\omega} \quad (11)$$

$$s_i^t = \left(a_{h,i} C_{h,i}^t \right)^T \quad s_i^t \in R^n \tag{12}$$

where h_i^t denotes the hidden state of the i -th node. ω is the size of the window for capturing the proximity time-slice, which determines how many hidden states from the previous time-steps are to be considered to construct the proximity state of the current node. $e_{h,i}^t$ is the hidden state of multiple time-slice of the i -th node that is mapped to a new vector by a mapping operation for the subsequent attentional weight computation. Q_h and r denote the parameters that are used to optimize the context-based attention mechanism parameters of the optimization based contextual attention mechanism. $a_{h,i}^t$ is a vector of attentional weights computed by softmax function in vector $e_{h,i}^t$. This attentional weight is used to represent the importance of the hidden states of different time slices.

$$S^t = \left(a_H C_H^t \right)^T \tag{13}$$

4.3. Dynamic Update Mechanism Based on GRU

In order to realize the dynamic detection of gantry anomalies, we need to capture the transaction characteristics of the gantry topology in different time slices, and the anomalies of the gantry will also change over time. GRU is a kind of Recurrent Neural Network (RNN) that solves the problems of Long Short Term Memory (LSTM) dependence and gradient vanishing in the back-propagation, and is similar to replacing the forgetting gates in LSTM. The input gates in LSTM are replaced by update gates, which makes the GRU network have fewer parameters and improves the training efficiency to a large extent. Therefore, in this paper, we use GRU to model the sequence changes, and capture the node information of the neighboring time slices to the current time slice, as well as the node long-term dependence information. The structure is shown in Figure 5.

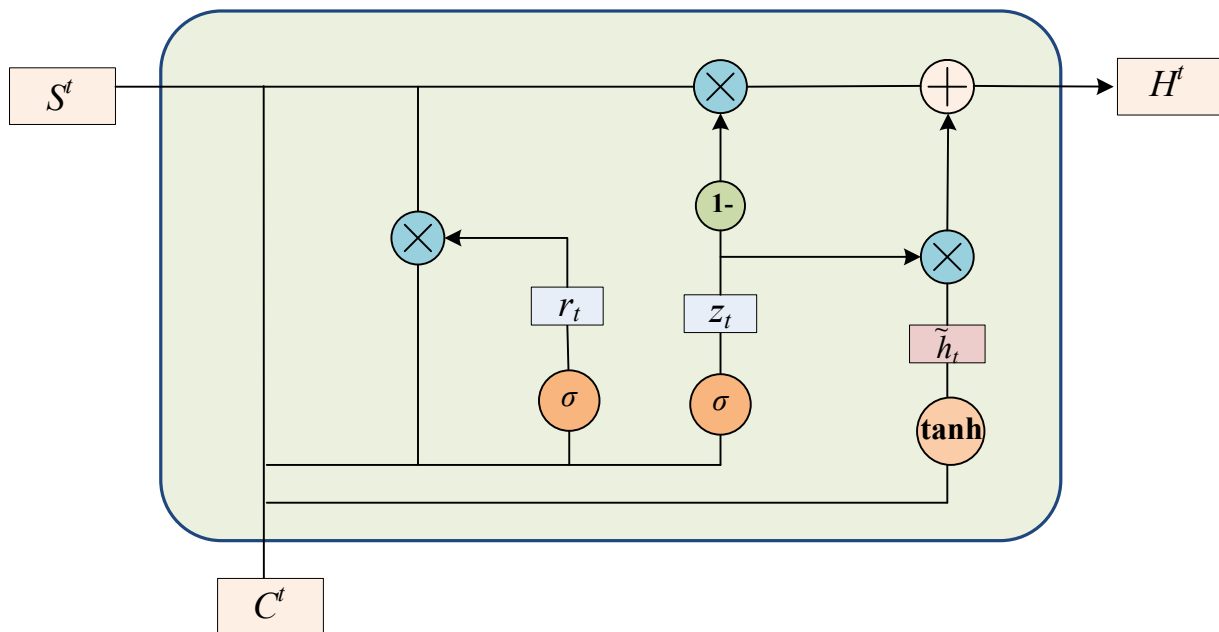


Figure 5. GRU schematic.

GRU contains two gating units: reset gate (r_t) can be combined with tanh to change the original information (upper position output and this position input) to obtain the result after processing the original information. Update gate (z_t) will contain the result after processing the reset gate selected by the update gate, and the weight of the upper position

output is summed up to obtain the current position output. The specific computation of the GRU process is as follows:

$$z_t = \sigma(U_z C^t + W_z S^t + b_z) \quad (14)$$

$$r^t = \sigma(U_r C^t + W_r S^t + b_r) \quad (15)$$

$$\tilde{H}^t = \sigma(U_c C^t + W_c (r^t \odot S^t)) \quad (16)$$

$$H^t = z_t \odot \tilde{H}^t + (1 - z_t) \odot S^t \quad (17)$$

where σ is the activation function \tanh , C^t is the state representation of the gantry node at the current time slice t , obtained by the GCN model aggregation in Section 4.1, S^t is the state representation of the node's approaching time window, obtained by the contextual attention mechanism in Section 4.2, Equation (16) denotes the current time-heavy storage information, and Equation (17) denotes the calculation of the update gates z_t for the current time, the hidden state of the approaching time window S^t , and the weighted average of the three current pieces of time-weighted storage information \tilde{H}^t . Finally, the hidden state matrix \tilde{H}^t of the node at timestamp t is computed.

Based on the above three subsections, we obtained the state matrix H^t for node t of the time slice and, according to the problem definition, we categorized the n node states in the matrix into five categories, N (normal), BA, MCA, MoCA, and SCA, i.e., $C = 5$, and computed the score for each node:

$$S_{i,j} = h_i^t \cdot W_j^T + b_j \quad (18)$$

where h_i^t is the hidden state of the i -th node, W is the category weight matrix $W \in R^{C \times d}$, W_j is the weight vector of the j -th category, and b_j is the bias term of the j -th category. for each node i , the Softmax function is used to compute the score on the category to which it belongs, and to obtain the probability distribution of each category.

5. Experimental Results and Analysis

5.1. Analysis of Gantry Transactions

In this paper, the ETC transaction data of 5,143,797 items of a provincial highway on 1 June 2021 were selected for analysis. This study focuses on the transactions mentioned in Definition 4, so redundant data, missing data, and erroneous data were eliminated. The following is an analysis of the transaction data for the 806 gantries on 1 June. Figure 6a shows the distribution of the different percentages of failed gantry transactions, and it can be seen that most of them are concentrated with a failure rate of 10% or less. In order to gain insight into the majority of gantry deal failures, this paper further analyzed the data with an anomaly rate of 10%. Figure 6b–d demonstrate the distribution of gantries with anomaly rates of 0–10%, 0–0.1%, and 0–0.01%, respectively, The share of failed gantry transactions is distributed at intervals of 0–10%, and 0–0.1%, with a decreasing trend in the number as the share increases. The number of gantries is more evenly distributed in the 0–0.01% interval compared to the other intervals.

Based on the above gantry transaction share scenarios, this paper further analyzes the gantries with high failure rates and those with low failure rates, respectively. First, we select nine gantries with transaction failure rates at five intervals of 1–5%, 5–10%, 10–20%, 20–50%, and 50–100%, whose transactions are shown in Figure 7. Then, a local topology containing these nine gantries was selected based on the standard topology of gantries. The local topology contains eight sections, and contains gantries with low failure rates that are connected to the nine gantries, which have failure rates between 0 and 0.1%. Finally,

the transaction data containing these eight road segments were filtered out of the 5 million ETC transaction data.

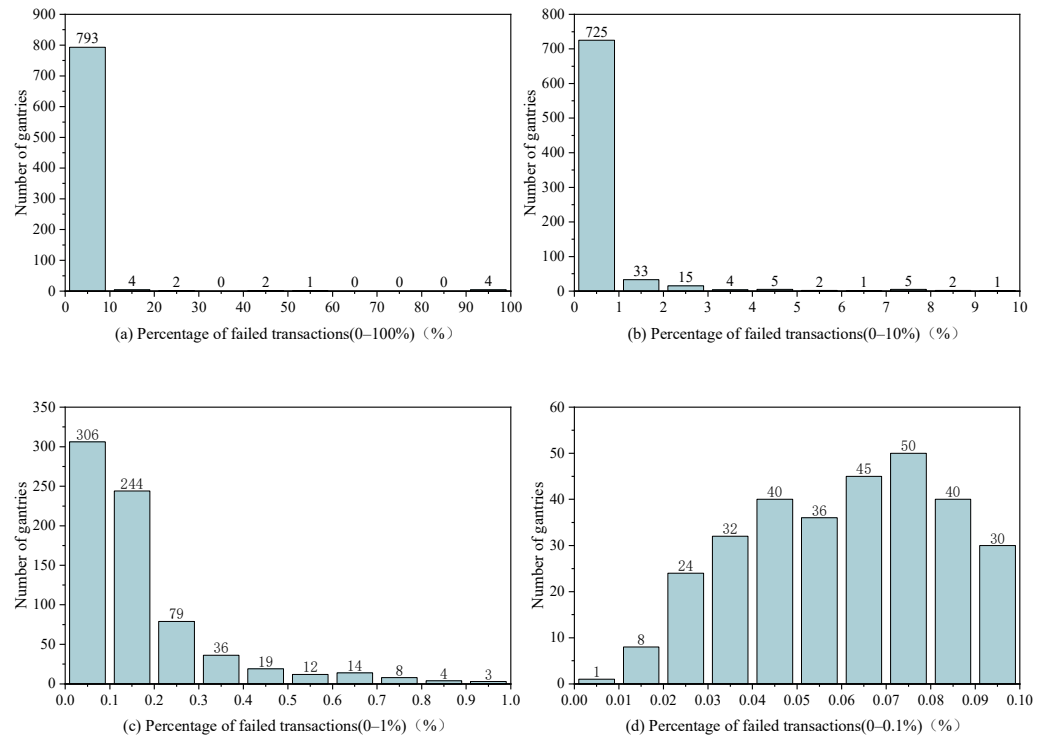


Figure 6. Analysis of the percentage of failed gantry transactions, (a) percentage of failed transactione (0–100%), (b) percentage of failed transactione (0–10%), (c) percentage of failed transactione (0–1%), (d) percentage of failed transactione (0–0.1%).

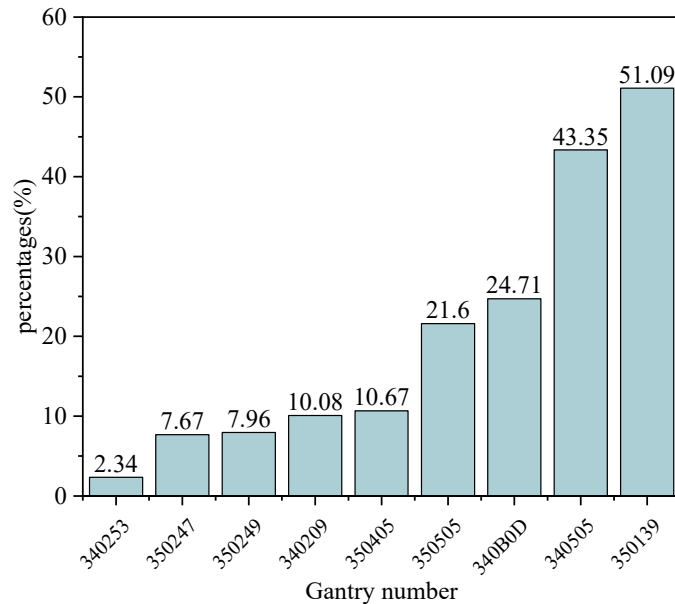


Figure 7. Percentage of failed gantry transactions.

The following paper analyzes the failure of gantry deals in these eight specific sections. The percentage of each type of anomaly for gantries with high failure rates is presented in Figure 8. These gantries mainly exhibit the four anomalies in Definition 5, with BA having the highest percentage of each of them. As the transaction failure rate increases, the proportion of SCA gradually increases and the proportion of BA gradually decreases. For

example, gantry 350,139 has the highest transaction failure rate in Figure 7, and in Figure 8, this gantry has the lowest percentage of BA and the highest percentage of SCA compared to other gantries.

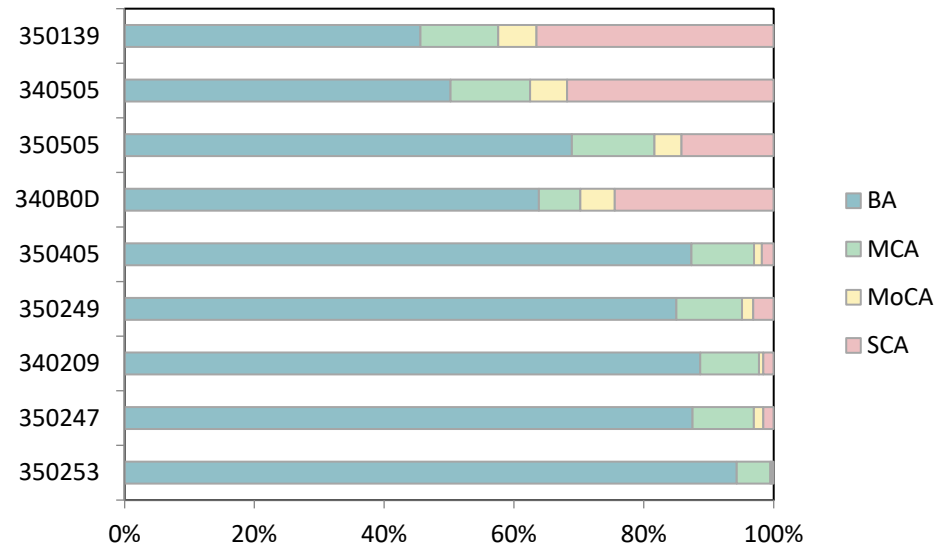


Figure 8. Percentage of different anomalies in high-failure-rate gantries.

An analysis of the sudden anomalous BA of the gantries with high failure rates was carried out as shown in Figure 9, which presents the transaction failures generated by different numbers of vehicles at different intervals. For gantry 350,139, with the highest transaction failure rate, the highest percentage of intervals 1 was found. The percentage of all types of intervals decreases as the intervals increase, indicating that the frequency of intervals leading from this gantry is high.

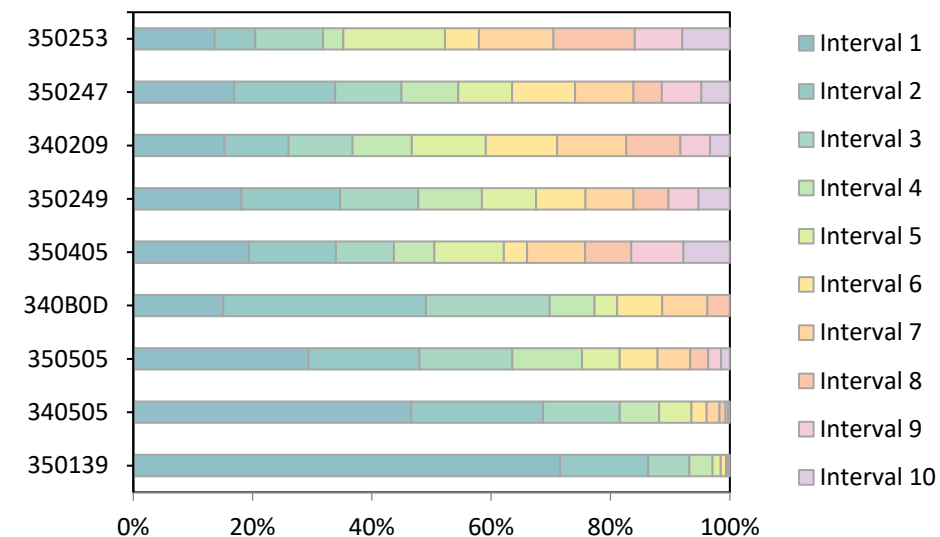


Figure 9. Analysis of BA in high-failure-rate gantries.

Similarly, five gantries with a failure rate of 0–0.1% were selected for analysis. These gantries do not have the three types of anomalies, MCA, MoCA, and SCA, and only contain BA anomalies. As shown in Figure 10, the number of intervals for these five low-anomaly-rate gantries is in the range of 20–200, and the frequency of sudden anomalies in these gantries is extremely low compared to the intervals of 1–10 in Figure 9.

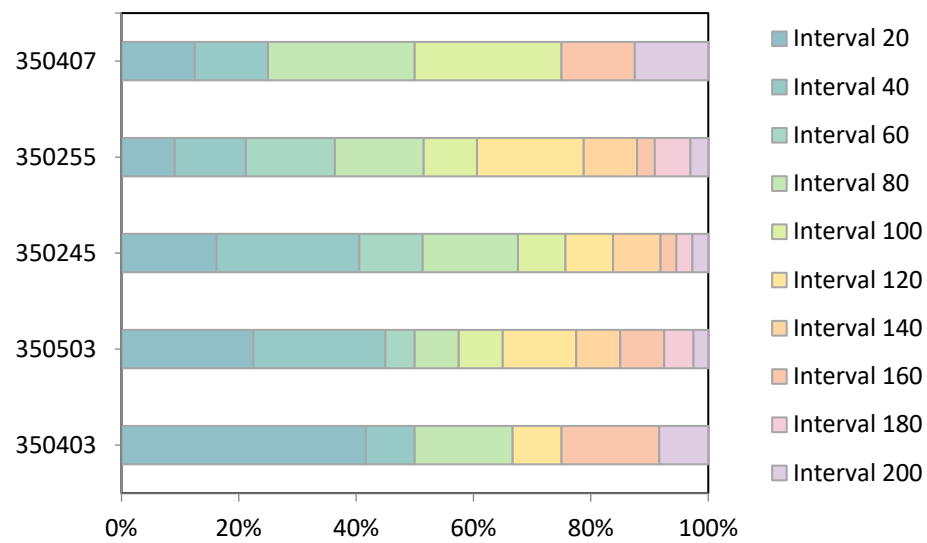


Figure 10. Analysis of BA in low-failure-rate gantries.

To summarize, the gantries with a high transaction failure rate will contain the four categories of anomalies defined in this paper, and the gantries with a low transaction failure rate indicate good operation, with only BA anomalies defined in this paper. The different categories can be graded to respond to the different operation status of the gantries.

5.2. Experimental Setup

In this paper, a 42 gantries’ transaction dataset from eight road sections on June 1 was divided into different node-number datasets, which were used to verify the model performance under different data volumes. Then, the same method as in the previous section was used to filter the relevant data of 42 gantries in nine road sections from 728,196 ETC transaction data on June 2, and this dataset contains 5000 nodes, which were used for the experiments comparing different models. We divided the training set and test set with an 8:2 ratio. Due to the large gap between the number of nodes of different categories in the dataset, we used stratified sampling to ensure that the number of nodes of each category in the training set is relatively balanced when the model is trained, and the model can better learn the features of each category, which improves the generalization ability of the model and reduces the risk of model overfitting for certain categories.

In order to validate the performance of the model, this paper uses classification accuracy (Acc), precision (Pre), recall (Rec) and F1-Score (F1) as evaluation metrics. The definitions are shown in Table 1, where TP, TN, FP and FN represent the number of true positives, true negatives, false positives and false negatives, respectively.

Table 1. Indicators for model evaluation.

Metric	Equation	Definition
Precision	$Pre = \frac{TP}{TP+FP}$	A measure of the proportion of samples predicted by the model to be in the positive category that are actually in the positive category.
Accuracy	$Acc = \frac{TP+TN}{TP+TN+FP+FN}$	A measure of the proportion of all samples that the model correctly classified.
Recall	$Rec = \frac{TP}{TP+FN}$	A measure of the proportion of all samples in which the model correctly predicted a positive class out of all samples that actually have a positive class.
F1-Score	$F1 = 2 \cdot \frac{Pre \cdot Rec}{Pre + Rec}$	A harmonic average combining precision and recall is used to comprehensively evaluate the performance of the model.

5.3. Experimental Analysis

5.3.1. Parameter Sensitivity

The experiments in this paper are based on the Pytorch deep learning architecture. The experimental learning rate was set to 1×10^{-1} , the weight decay was set to 5×10^{-4} , and the maximum number of training times for the experiment was 400. In order to determine influence of the total number of layers L and the hidden state on the overall model in GCN, this paper selected the range of L as $\{1, 2, 3, 4, 5\}$ and the range of d as $\{4, 8, 16, 32, 64, 128\}$, and set the other parameters to be optimal. As shown in Figure 11, the four plots represent the values of accuracy, precision, recall and F1-Score at different L, d values. The overall contours of the four plots are generally consistent. The values of each evaluation metric gradually increase when L increases from 1 to 2, and L undergoes a slight decrease from 2 to 3, so that it reaches its peak when L is 2. As d increases, the value of each evaluation metric fluctuates and peaks at d of 4 and 32, but reaches the optimal value at d of 32. After d and L reach the optimal configuration, the value of each evaluation metric decreases as it continues to increase. When there are too many layers, the GCN may capture useless information about remote neighbors, which reduces the accuracy of the framework. A large d increases the complexity of the framework and makes it more difficult to converge to the optimal point.

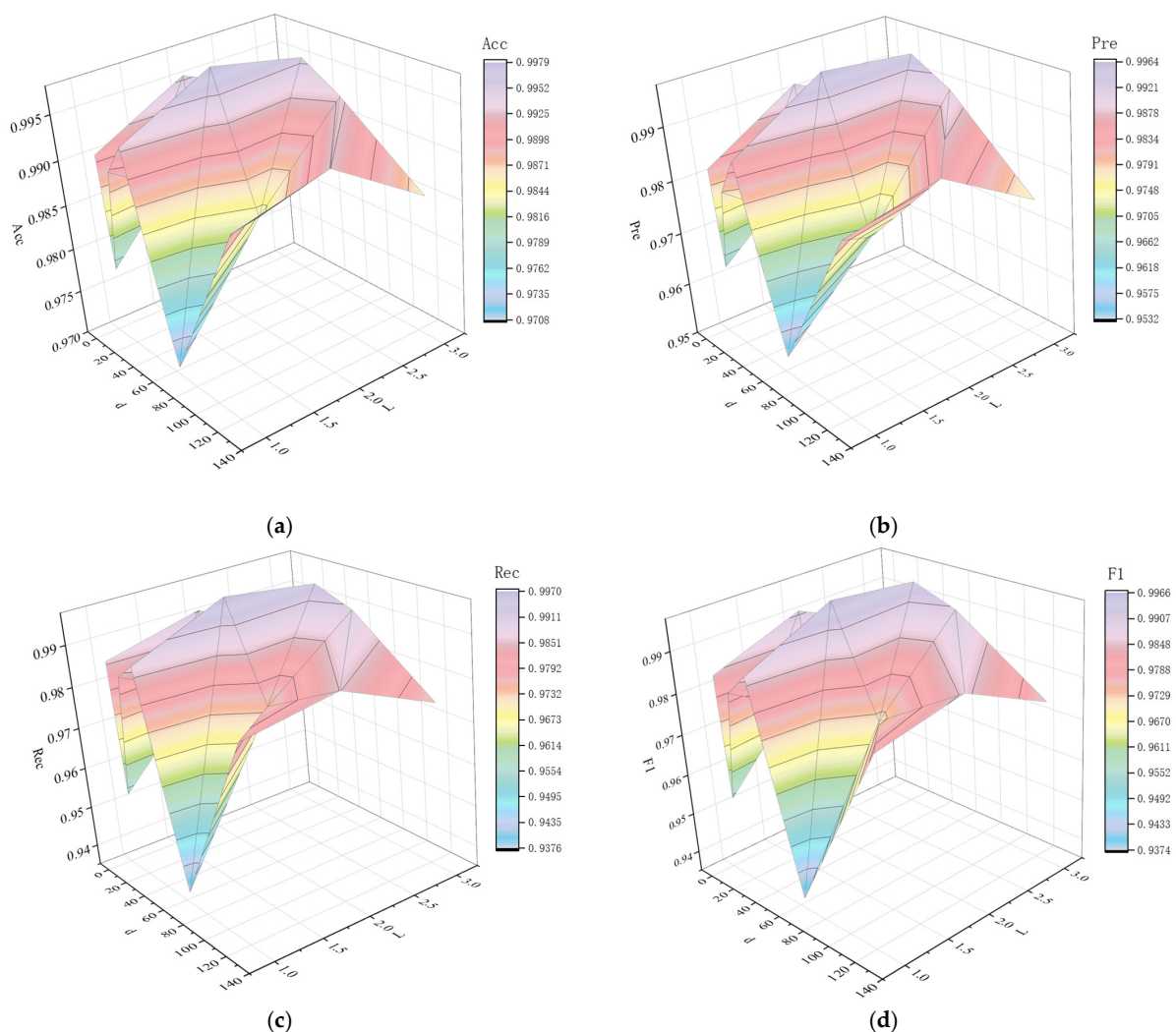


Figure 11. Results for evaluation indicators with different parameters: (a) results for Acc with different parameters, (b) results on Pre with different parameters, (c) results for Rec with different parameters, (d) results for F1 with different parameters.

5.3.2. Comparison Experiment

We compared the model in this paper with three groups of baselines. The first group considers only node features, including MLP, XGBoost, and SVM [30]. The second group contains generalized GNN models, including GCN [31], GAT [32], and GraphSAGE [33]. The third group comprises the combined model GCN-LSTM considering spatio-temporal features.

- MLP: a multi-layer perceptron network consisting of two linear layers with activation functions.
- XGBoost: a gradient boosting ensemble algorithm that leverages a collection of decision trees as its base learners, employing sophisticated optimization techniques to enhance predictive accuracy and handle a wide range of machine learning tasks.
- SVM: a support vector machine with the Radial Basis Function (RBF) kernel.
- GCN: a graph convolutional network using the first-order approximation of localized spectral filters on graphs.
- GAT: a graph attention network that employs the attention mechanism for neighbor aggregation.
- GraphSAGE: a GNN model based on a fixed sample number of the neighbor nodes.
- GCN-LSTM: a hybrid deep learning architecture that combines the power of graph convolutional networks (GCNs) with the sequential modeling abilities of Long Short-Term Memory networks (LSTMs).

The results of the comparison of different models are shown in Table 2 and Figure 12. The figure visualizes that this paper's model GCN-GRU-Attention achieves the best performance in all evaluation metrics, which can be seen in the table with values above 99%, and in the generalized GNN model, GAT performs poorly in most cases, with values around 70%, except for the accuracy. MLP, XGBoost, and SVM also achieve a comparable performance when graph structure is ignored, with metrics values of around 80% for each series, the models GraphSAGE and the combined model GCN-LSTM achieve relatively better results, with metrics values of around 95% for each series, and our model achieves a better performance in terms of accuracy, precision, recall, and F1 values compared to the GCN-LSTM, by 0.99%, 1.69%, 1.89%, and 1.93%, respectively. This also shows that ETC transaction data have graph structure characteristics as well as temporal characteristics, and the model chosen in this paper is very suitable for gantry anomaly detection based on ETC transaction data.

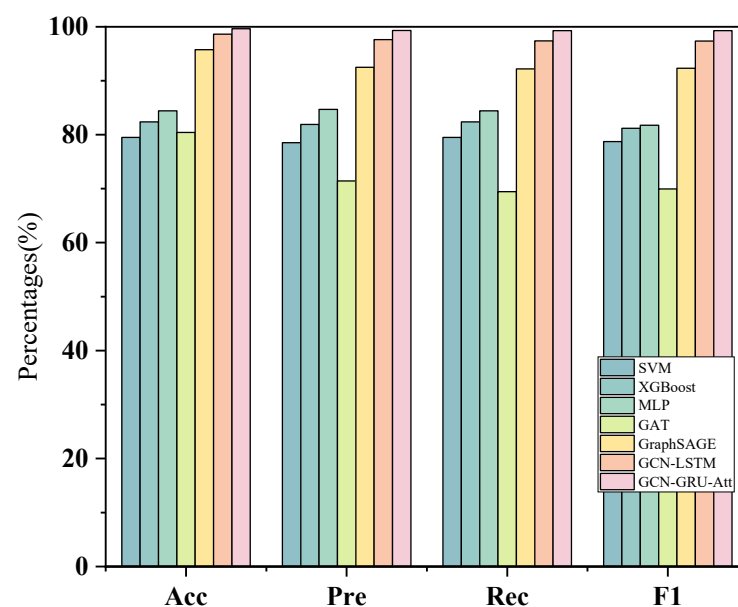


Figure 12. Performance comparison of different models.

Table 2. Performance comparison of different models.

Model	Acc (%)	Pre (%)	Rec (%)	F1
SVM	79.51	78.52	79.51	78.72
XGBoost	82.36	81.92	82.36	81.19
MLP	84.43	84.67	84.43	81.74
GAT	80.43	71.43	69.46	69.94
GraphSAGE	95.76	92.49	92.19	92.32
GCN-LSTM	98.62	97.61	97.37	97.35
GCN-GRU-Attention	99.61	99.30	99.26	99.28

5.3.3. Ablation Experiment

For the different model components (GCN, GRU, Attention), this paper uses ablation experiments to quantitatively assess the importance of each component in the model. As shown in Table 3, each row represents a different model configuration, where the GCN updates the representation of each node through the propagation of information from neighboring nodes, is able to take into account the relationship between a node and its neighbors, and is to be used to process static graph data. The GRU is dedicated to sequence data and is suitable for processing dynamic graph data, with the ability to model temporal information. The attention mechanism allows for the model to assign different attention weights according to different parts of the input sequence, and the model can more effectively select information that is relevant to the current task, thus improving its performance.

Table 3. Model ablation experiment performance table.

GCN	GRU	Attention	Acc (%)	Pre (%)	Rec (%)	F1
√	×	×	87.12	88.24	87.12	86.59
×	√	×	81.99	81.60	81.99	80.47
√	√	×	93.44	92.00	85.80	83.99
√	√	√	99.61	99.30	99.26	99.28

The gantry node transaction data studied in this paper are temporal and spatial in nature; therefore, the experimental results show that the performance of the model using only a single technique is not as good as the model integrating multiple techniques. The metrics of precision, accuracy and F1 score show that the model that uses GCN, GRU and Attention at the same time has the best results, with accuracy and F1 scores reaching over 99%, which is significantly better than the other three models. This indicates that the combination of GCN, GRU and Attention has a significant, positive impact on improving model performance in this experiment. The combination of GCN and GRU significantly improves the performance compared to the use of GCN or GRU alone, indicating that the combination of the two can extract more effective features and enhance the performance of the model. In addition, the introduction of the Attention mechanism can further improve the performance of the model, because the Attention mechanism makes the model better able to handle long-distance dependencies in sequences.

5.3.4. Model Performance Analysis

In order to verify the training performance of this paper's model under different data volumes, four datasets were selected in this paper, with 1000, 2000, 3000, and 5000 nodes, respectively. As shown in Figure 13, in datasets 1 and 2, the training loss values and accuracy rates of the two datasets change dramatically when the epoch is less than 50, and show a small range of fluctuation after 50 epochs, at which point the training accuracy tends to increase and training loss value tends to decrease. When the epoch is between 300 and 400, the accuracy fluctuates slightly at around 0.97–0.98, and the loss value fluctuates slightly at around 0.13, and then stabilizes. In datasets 3 and 4, the training loss values and

accuracy of both datasets change drastically when the epoch is less than 50, and fluctuate more dramatically when the epoch is in the interval range from 50 to 200, but fluctuate less after the epoch is greater than 200. When the epoch is between 300 and 400, the accuracy fluctuates around 98%, and the loss value fluctuates around 0.12 and then also stabilizes around 400. This shows that the trained network is able to adapt to new unseen data with the same distribution as the training data, and that the size of the dataset does not have a significant effect on the convergence of the loss values of the model and the accuracy of the model results.

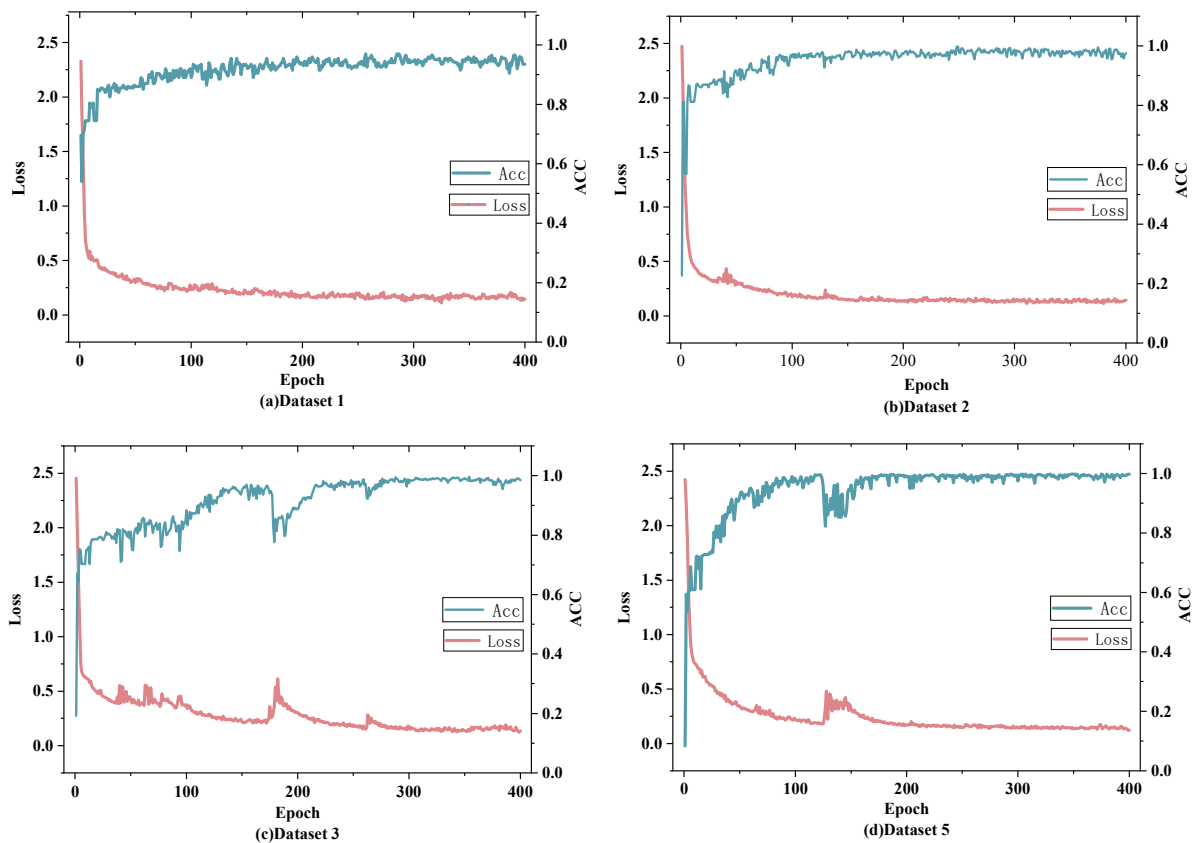


Figure 13. Training performance of the model: (a) training performance with 1000 nodes, (b) training performance with 2000 nodes, (c) training performance with 3000 nodes, (d) training performance with 5000 nodes.

In order to further verify the changes in each evaluation metric (Acc(%), Pre(%), Rec(%), F1) of the model under different data volumes, as shown in Figure 14, the horizontal coordinate is the number of different nodes and the vertical coordinate is the value of the evaluation metrics. Overall, as the data volume increases, the value of each indicator gradually increases, and finally each value reaches the optimum when the number of nodes is 5000. Although the size of the data volume has an impact on the performance of the model, the impact is not large. When 1000 nodes are used in small datasets, in this paper's model with an Acc value of about 96%, the lower F1 value is also in the range of 89% or so; when 5000 are used in a large dataset, the model of the value of each evaluation index is in the range of 99% or so. In summary, the effectiveness of this paper's algorithmic model in terms of spatio-temporal data is proved, and the robustness of retrieval is guaranteed. Additionally, the model has a good generalization ability.

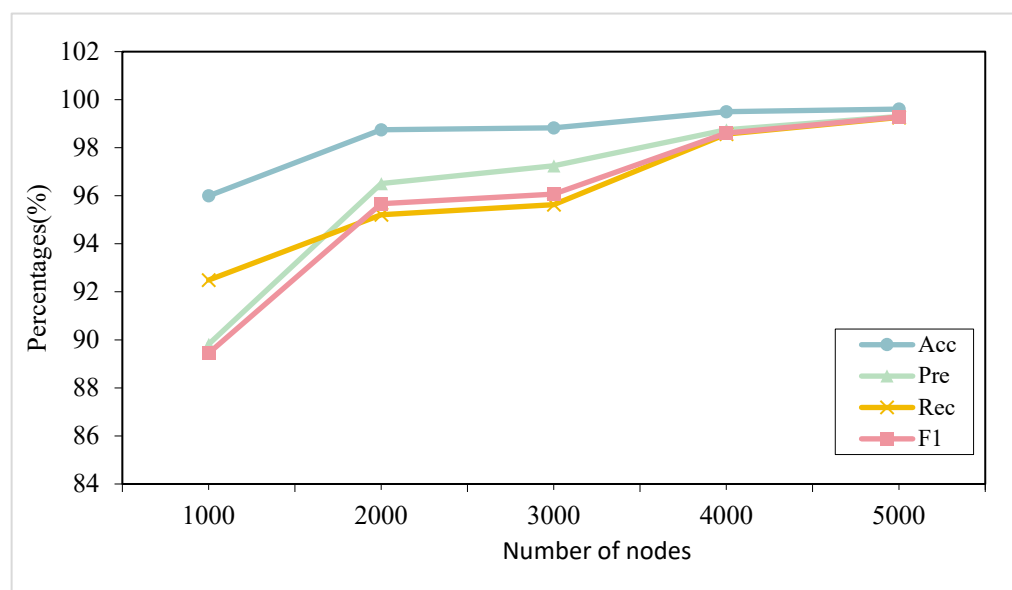


Figure 14. Changes in evaluation indicators at different data volumes.

6. Conclusions

Dynamic graph anomaly detection has a relatively important research status and practical significance. For the first time, we introduced gantry transaction data for gantry anomaly detection, an innovative approach that allows for us to comprehensively assess gantry anomalies. This classification system provides a new framework for gantry equipment management, allowing for operations and maintenance personnel to gain a more comprehensive understanding of gantry system problems and develop more targeted maintenance strategies. We also employed a composite model including GCN, context-focused mechanisms, and GRU to comprehensively consider the spatial and temporal characteristics of gantry data. This approach demonstrates excellent performance in experiments, with over 99% accuracy, a 2–20% performance improvement over other traditional anomaly detection algorithms and other graph flow anomaly detection methods, and the fluctuation range of the model's performance metrics is within 10% under different data volumes. This finding provides strong support for further research and its practical applications in the field of dynamic graph anomaly detection. Finally, despite the remarkable results of our research, there are still some limitations. Our method is based on offline data, and thus presents challenges in terms of real-time performance. Future research directions will include the establishment of a simulation system for the dynamic detection of real-time gantry transaction data to meet the practical needs of gantry equipment operation and maintenance.

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

Funding: This work is partially supported by the Renewable Energy Technology Research institution of Fujian University of Technology Ningde, China (Funding number: KY310338, Funder: F.Z.), the 2020 Fujian Province “Belt and Road” Technology Innovation Platform (Funding number: 2020D002, Funder: F.Z.), the Provincial Candidates for the Hundred, Thousand and Ten Thousand Talent of Fujian (Funding number: GY-Z19113, Funder: F.Z.), the Patent Grant project (Funding number: GY-Z18081, GY-Z19099, GY-Z20074, Funder: F.Z.), Horizontal projects (Funding number: GY-H-20077, Funder: F.Z.), Municipal level science and technology projects (Funding number: GY-Z-22006, GY-Z-220230, Funder: F.Z.), Fujian Provincial Department of Science and Technology Foreign

Cooperation Project (Funding number: 2023I0024, Funder: F.Z.), the Open Fund project (Funding number: KF-X19002, KF-19-22001, Funder: F.Z.).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The ETC transaction data utilized in this study were obtained from Fujian Expressway Information Technology Co., Ltd., Fu Zhou, China. Restrictions apply to the availability of these data, which were used under license for this study and are not publicly available. Data are available from the authors with the permission of Fujian Expressway Information Technology Co., Ltd. All data processing and analyses were conducted in compliance with relevant data protection and privacy laws. No individual or personal data were used in this study.

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

References

1. Du, Y.C.; Liu, C.L.; Wu, D.F.; Hao, C.Z. Framework of the New Generation of Smart Highway. *China J. Highw. Transp.* **2022**, *35*, 203–214.
2. Qian, M. Analysis of multi-dimensional data fusion and application of ETC portal system. *China ITS J.* **2021**, *6*, 109–112.
3. Guo, F.; Zou, F.; Luo, S.; Liao, L.; Wu, J.; Yu, X.; Zhang, C. The Fast detection of Abnormal ETC Data Based on an Improved DTW Algorithm. *Electronics* **2022**, *11*, 1981. [[CrossRef](#)]
4. Alexakis, T.; Peppes, N.; Demestichas, K.; Adamopoulou, E. A distributed big data analytics architecture for vehicle sensor data. *Sensors* **2022**, *23*, 357. [[CrossRef](#)] [[PubMed](#)]
5. Yinglei, H.; Dexin, Q.; Shengyuan, Z. Smart transportation travel model based on multiple data sources fusion for defense systems. *Soft Comput.* **2022**, *26*, 3247–3259. [[CrossRef](#)]
6. Liu, Y.; Cai, Z.; Dou, H. Highway traffic congestion detection and evaluation based on deep learning techniques. *Soft Comput.* **2023**, *27*, 12249–12265. [[CrossRef](#)]
7. Niu, J.; He, J.; Li, Y.; Zhang, S. Highway Temporal-Spatial Traffic Flow Performance Estimation by Using Gantry Toll Collection Samples: A Deep Learning Method. *Math. Probl. Eng.* **2022**, *2022*, 8711567. [[CrossRef](#)]
8. Zheng, Z.E.; Zhou, J.; Dai, J.J. Intelligent Research and Application of Highway ETC Gantry System. *Hu Nan Commun. Sci. Technol.* **2021**, *47*, 48–52.
9. Jing, B.; Zhuang, N.C. ETC gantry detection and operation and maintenance management discussion. *China ITS J.* **2020**, *12*, 39–41.
10. Wei, K.X.; Li, J.G. Cloud Alarm System for Key Equipment of Highway Toll Collection System. *China ITS J.* **2021**, *262*, 127–129.
11. Kuncan, M.; Kaplan, K.; Minaz, M.R.; Kaya, Y.; Ertunç, H.M. A novel feature extraction method for bearing fault classification with one dimensional ternary patterns. *ISA Trans.* **2020**, *100*, 46–57. [[CrossRef](#)] [[PubMed](#)]
12. Li, Y.; Ding, K.; He, G.; Jiao, X. Non-stationary vibration feature extraction method based on sparse decomposition and order tracking for gearbox fault diagnosis. *Measurement* **2018**, *124*, 53–69. [[CrossRef](#)]
13. Zhang, S.; Tang, J. Integrating angle-frequency domain synchronous averaging technique with feature extraction for gear fault diagnosis. *Mech. Syst. Signal Process.* **2018**, *99*, 711–729. [[CrossRef](#)]
14. Qian, W.; Li, S.; Wang, J.; Wu, Q. A novel supervised sparse feature extraction method and its application on rotating machine fault diagnosis. *Neuro Comput.* **2018**, *320*, 129–140. [[CrossRef](#)]
15. Li, W.; Qiu, M.; Zhu, Z.; Wu, B.; Zhou, G. Bearing fault diagnosis based on spectrum images of vibration signals. *Meas. Sci. Technol.* **2016**, *27*, 035005. [[CrossRef](#)]
16. Liang, Y.; Zhang, Y.Y.; Ming, Y.G. Fault detection method based on generalized mutual entropy principal element analysis. *J. Taiyuan Univ. Technol.* **2020**, *3*, 438–445.
17. Shao, Y.; Wu, J.W.; Liang, M.S. An Integrated SVM Approach under AM-ReliefF Feature Selection for Mechanical Fault Diagnosis of High Voltage Circuit Breakers. *Proc. CSEE* **2021**, *41*, 2890–2901.
18. Yang, J.; Wan, P.A.; Lin, W.J. Aero-engine bearing fault diagnosis based on multi-sensor fusion convolutional neural network. *J. Zhejiang Univ.* **2022**, *42*, 4933–4942.
19. Liu, J. Real-Time Abnormal Detection of Train Operation Speed of Urban Rail Transit Based on XGBoost Model. *J. Chongqing Jiaotong Univ.* **2021**, *40*, 49.
20. Ranshous, S.; Harenberg, S.; Sharma, K.; Samatova, N.F. A scalable approach for outlier detection in edge streams using sketch-based approximations. In Proceedings of the 2016 SIAM Int'l Conference on Data Mining (SDM), Miami, FL, USA, 5–7 May 2016; pp. 189–197.
21. Eswaran, D.; Faloutsos, C. Sedanspot: Detecting anomalies in edge streams. In Proceedings of the Int'l Conference on Data Mining (ICDM), Singapore, 17–20 November 2018; pp. 953–958.
22. Zhang, R.Z. *Anomalous Node Detection in Social Networks Based on Graph Neural Networks*; Southwest Jiaotong University: Chengdu, China, 2021.
23. Huyan, K.; Fan, X.; Yu, L.T.; Luo, Z.X. Graph based neural network regression strategy for facial image superresolution. *J. Softw.* **2018**, *29*, 914–925.

24. Qu, Q.; Yu, H.T.; Huang, R.Y. Graph convolutional network based social network Spammer detection technology. *Chin. J. Netw. Inf. Secur.* **2018**, *30*, 43–50.
25. Ning, S.Q.; Guo, M.Z.; Ren, S.J. A semi-supervised method for cancer clinical outcome prediction based on graph convolution network. *Intell. Comput. Appl.* **2018**, *8*, 44–48.
26. Liu, Y.; Pan, S.; Wang, Y.G. Anomaly detection in dynamic graphs via transformer. *IEEE Trans. Knowl. Data Eng.* **2021**, *14*, 8. [[CrossRef](#)]
27. Ding, Q.; Li, J. AnoGLA: An efficient scheme to improve network anomaly detection. *J. Inf. Secur. Appl.* **2022**, *66*, 103149. [[CrossRef](#)]
28. Zou, F.; Ren, Q.; Tian, J.; Guo, F.; Huang, S.; Liao, L.; Wu, J. Expressway Speed Prediction Based on Electronic Toll Collection Data. *Electronics* **2022**, *11*, 1613. [[CrossRef](#)]
29. Sankar, A.; Wu, Y.; Gou, L.; Zhang, W.; Yang, H. Dysat: Deep neural representation learning on dynamic graphs via self-attention networks. In Proceedings of the 13th International Conference on Web Search and Data Mining, Houston, TX, USA, 3–7 February 2020; pp. 519–527.
30. Chang, C.C.; Lin, C.J. A library for support vector machines. *ACM Trans. Intell. Syst. Technol. (TIST)* **2011**, *2*, 1–27. [[CrossRef](#)]
31. Kipf, T.N.; Welling, M. Semi-supervised classification with graph convolutional networks. In Proceedings of the ICLR, Toulon, France, 24–26 April 2017.
32. Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Lio, P.; Bengio, Y. Graph attention networks. *arXiv* **2017**, arXiv:1710.10903.
33. Hamilton, W.; Ying, Z.; Leskovec, J. Inductive representation learning on large graphs. In Proceedings of the NeurIPS, Long Beach, CA, USA, 4–9 December 2017; pp. 1025–1035.

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.