Association Rule Mining and Node-Level Fault Prediction in Vehicles Using Graph Neural Networks (NRP-GCN)

by

You WU 2130026160

Sirui ZHANG 2130026199

Junwei SU 2130026129

A Final Year Project Thesis
submitted in partial fulfillment of the requirements
for the degree of

Bachelor of Science
in
Data Science
at
BNU-HKBU
UNITED INTERNATIONAL COLLEGE

November, 2024

ABSTRACT

The accurate prediction of faults in vehicle chassis systems is very crucial for guaranteeing reliability and safety in the automotive industry. Modern vehicles produce a huge volume of sensor data, which challenges traditional fault prediction methods due to the complexity, diversity, scale, and key spatial-temporal relations embedded in the data. Besides, current models often cannot generalize well across different types of chassis.

In this line of challenges, this paper proposes a Node-Level Risk Prediction Graph Convolutional Network for its application in fault prediction on vehicle chassis systems. The main objective of this paper is to enhance the accuracy of fault diagnosis by allowing the model to learn across different chassis types, so it can find fault patterns across highly variant vehicle models. This model will have the ability to embed better spatial correlations in sensor data through GNNs than could ever be allowed by traditional machine learning methodologies.

Our model significantly outperforms conventional methods, as demonstrated by its superior performance in predicting three critical target categories: `EmergencyFeeCategory`, `HasEmergencyFee`, and `FaultCategory`. These categories reflect key aspects of vehicle faults, such as the presence of emergency fees, the type of emergency fee, and the specific fault category. The proposed NRP-GCN approach effectively models the dependencies between these categories, leading to more accurate predictions and earlier fault detection.

Herein, each vehicle chassis is represented as a node in the graph. The edges characterize the relationships between the components, guided by operational data and features of faults. The graphical representation model can learn the complicated fault patterns from various vehicles. In this paper, an NRP-GCN is proposed, which is designed to effectively handle heterogeneous sensor data. It should be reliable under a variety of real-world scenarios.

The experimental results have shown that NRP-GCN is far superior to traditional machine learning models in the accuracy of its prediction performance for all three target

categories, showing their improvement in accuracy while possessing very strong generalization capability, hence making this model very practically valuable under real traffic conditions. It will be helpful for early fault detection and hence will help bound the occurrence of unexpected failures, improving vehicle safety and efficiency of maintenance.

Contents

DECLARATION

I hereby declare that all the work done in this project is of my independent effort. I also certify that I have never submitted the idea and product of this project for any academic or employment credits.

Signature: ______________________________

Student Name: You WU, Sirui ZHANG, Junwei SU

Student ID: 2130026160, 2130026199, 2130026129

Date: November 19, 2024

BNU-HKBU
United International College
Data Science Program

We hereby recommend that the project submitted by student **You WU, Sirui ZHANG, Junwei** SU, entitled **“Association Rule Mining and Node-Level Fault Prediction in Vehicles Using Graph Neural Networks (NRP-GCN)”**, is accepted in partial fulfillment of the requirements for the degree of Bachelor of Science (Honours) in Data Science Program.

Date:

Date:

ACKNOWLEDGEMENT

We would like to extend our heartfelt gratitude to Dr. Zhijian Li for his exceptional guidance and support throughout our Final Year Project. His profound insights and thoughtful direction significantly shaped our approach and deepened our understanding of the subject matter.

Throughout the course of our project, Dr.Li was always available to offer positive and constructive feedback. Whether we were grappling with complex theories or technical challenges, he provided invaluable advice and solutions that helped us navigate obstacles with confidence. His patience in explaining difficult concepts and his willingness to engage in thorough discussions greatly enhanced our problem-solving skills.

Dr. Li's dedication to our success went beyond academic guidance. He encouraged us to think critically and innovatively, pushing us to explore new perspectives and methodologies. His encouragement fostered a supportive and stimulating learning environment, enabling us to grow both academically and personally. We are truly thankful for his mentorship, which has been invaluable in making this journey both enlightening and memorable.

You WU, Sirui ZHANG, Junwei SU
November 20, 2024

Chapter 1

Introduction

Fault diagnosis and prognosis are two very important activities in commercial vehicles for ensuring safety, efficiency, and reliability in a vehicle fleet. These kinds of vehicles operate everything from long hauls of freight to public transportation within conurbation limits, so unplanned failures will result in severe delays to operations. For example, a fault in the logistics fleet could delay the delivery and cost financially, besides dissatisfaction among customers; a fault in a public bus may turn out to be very dangerous with respect to passenger safety. The early prediction of the fault allows for quicker repairs and avoids costly breakdowns, reducing vehicle downtime hence increasing the general effectiveness of the fleets.

Fault observation involves one of the most important aspects: observing real-time mechanical and electronic systems. Modern-day vehicles consist of numerous sensors that monitor performance, braking, fuel efficiency, and so forth. A large volume of operational data is produced by these sensors; this data includes measures such as engine torque, coolant temperature, and the composition of gases in exhaust. But the problem remains at the level of how such information is analyzed and interpreted to understand the early signs of deterioration or an imminent failure; for example, abnormally high engine temperatures or sudden losses in fuel economy may hint at engine misfire or transmission failure.

Fault diagnosis is very critical as it forms a basis for active maintenance policies, hence minimizing cost of repairs and idle time. By detecting and solving problems before they evolve into more catastrophic failures, the fleet manager may avoid the financial consequences of unscheduled stops and maintain operational reliability. For example, early detection of defects in engines could avoid costly repairs, while the timely detection of braking system faults could improve road safety. This will further enhance economic efficiency while maintaining a high level of safety among passengers, cargo, and operators themselves.

However, existing methods for fault detection are severely limited. Much of such methods have been carried out under laboratory-generated or simulated conditions that cannot capture the complexity of real operational environments. Besides, traditional methods usually lead to losses in real-life applications due to a lack of representativeness regarding the training datasets and an inability to model complex relationships among the characteristics of faults. Due to such deficiencies, there is an increasingly high demand for advanced systems to handle diverse and realistic data sources more efficiently.

In this paper, we propose the Node-Level Risk Prediction Graph Neural Network (NRP-GCN) to address challenges in fault diagnosis using cross-chassis learning. Our approach begins with K-means clustering to group chassis based on spatial features, enabling the detection of common fault patterns across vehicles. Feature extraction is performed using Empirical Mode Decomposition (EMD) to simplify complex signals and enhance interpretability. The GNN architecture integrates spatial and operational features, with a Fourier-based graph convolution layer capturing complex spatial dependencies by transforming data into the frequency domain for analysis. Transfer learning further improves robustness by sharing knowledge among chassis with similar fault patterns. This holistic approach provides a reliable and proactive solution for vehicle fault prediction and maintenance.

Chapter 2

Related Work

In the field of fault prediction, the acquisition of anomalous events is often difficult, and current research typically treats anomaly detection as an unsupervised problem. Existing research tends to focus on single fault type prediction and anomaly detection based on fixed thresholds, involving the design of models that describe normal, non-anomalous data. Learned models are then used to detect anomalies by generating high scores to indicate abnormal events.

Compared to the time series data used in [1] (e.g., panel temperature and operating voltage) and the power plant data in [2], which contain partially real data and partially injected anomalies with limited data size, our fault data comes from real-world commercial vehicle operations. This gives it higher credibility and generalizability, allowing it to more accurately reflect the complexity and noise of real-world scenarios. Furthermore, we not only have detailed time series data to capture the dynamic changes before and after vehicle faults and analyze long-term trends, but also spatial features (e.g., GPS location information), enhancing our ability to analyze the relationship between geographical locations and faults. The combination of spatial and temporal data aids in constructing more accurate fault prediction models, identifying vehicle fault patterns under different environments, and providing more precise regional fault warnings.

Traditional analytical tools such as Support Vector Regression (SVR)[12], Gradient

Boosting Decision Trees (GBDT)(SVR)[13], Vector Autoregression (VAR) (SVR)[14], and Autoregressive Integrated Moving Average (ARIMA) (SVR)[15] often face challenges when dealing with complex temporal relationships. These conventional methods struggle to capture non-linear patterns and intricate interactions between variables in time series data, resulting in suboptimal predictive accuracy (SVR)[16]. With the advancement of deep learning techniques, neural network models based on Convolutional Neural Networks (CNN) (SVR)[17], Recurrent Neural Networks (RNN) (SVR)[18], and Transformers (SVR)[19] have emerged as leading approaches for handling time series data. These deep learning models have demonstrated significant advantages in modeling real-world, complex time series by effectively identifying intricate patterns and dependencies. However, despite their strengths, these methods exhibit a major limitation: they fail to explicitly model the spatial relationships between time series in non-Euclidean spaces (SVR)[20]. This shortcoming restricts their expressiveness and effectiveness in scenarios where time series data exhibit complex spatial structures.

For example, [3] primarily utilized deep learning and attention mechanisms to predict the remaining useful life (RUL) of turbofan engines, aiming to accurately estimate when the system would fail. However, its limitation lies in the lack of adaptability to multi-fault scenarios. [4] applied Temporal Convolutional Networks (TCN) for real-time fault detection and warning on vehicle sensor data but relied on fixed anomaly detection thresholds, making it difficult to adapt dynamically to different operational environments, leading to potential false positives or missed alarms. Similarly, [5] and [6] both relied on Support Vector Machines (SVM) for fault detection and prediction. While SVMs are advantageous for small-scale data, they demonstrate limitations in handling complex, high-dimensional sensor data. The scalability and real-time processing capability of these models are affected, and they typically cover a limited range of faults.

Additionally, [7] detected known and unknown faults using an ensemble-based anomaly detection approach, which improved the robustness of fault detection. However, the computational complexity was high when handling large-scale data, making real-time application challenging. Unlike these methods, [8] used deep neural networks for unsupervised anomaly detection, which can handle multivariate time series data. However, its limitation lies in the reduced reliance on label information, making it difficult to interpret specific reasons, and the model's complexity increases computational costs. [9] and [10] emphasized using statistical methods, such as Principal Component Analysis (PCA) and multivariate regression models, for detecting engine faults. While these methods excel at detecting weak fault signals, they rely heavily on large amounts of healthy data for modeling, making them less flexible in real-time applications and unable to adapt dynamically to complex environments.

Compared to the existing research, our current work appears to enjoy some apparent advantages and novelties: Our work does not develop a model for fault type detection alone but proposes a model using multi-task learning to predict fault types on multiple aspects, improving both generalization capability and applicability of models. A cross-platform learning framework was then proposed to solve the problem of data silos, allowing knowledge transfer across different vehicles, especially those with insufficient data, in order to enhance model robustness. We also introduce a dynamic adaptive threshold adjustment mechanism to avoid the limitations of traditional fixed-threshold methods, enabling the model to dynamically adjust the fault detection threshold with changes in vehicle operating conditions and external environments such as weather and road conditions to reduce false positives and missed alarms.

Meanwhile, a framework integrated with graph neural networks was developed to capture the spatial dependencies of vehicle sensor readings for improving fault prediction in complex scenarios. This framework uses GNNs to extract both relational and sequential patterns from graph-structured data and integrates these insights into tabular features using conventional machine learning modeling. Thus, integrating it achieves the accuracy of early fault detection. Real-time processing, adaptability of cross-platform learning, multi-task scalability, and dynamic threshold adjustment for our proposed model show remarkable advantages compared with traditional approaches.

Chapter 3

Research Methodology

3.1 Data Description

This section outlines the methodology used for fault detection and diagnosis, including data preprocessing, feature selection, Graph Neural Networks (GNNs), and time series analysis.

3.1.1 Fault Data

The fault data analyzed spans February to April 2023, comprising approximately 9 million records per month. These records represent over 4,000 fault types, with more than 5,000 unique fault types identified. This extensive dataset forms the foundation for fault detection and diagnosis.

Figure 3.1 illustrates the geographic distribution of faults along with associated first aid cost categories. Each point represents the average latitude and longitude location of a fault, with the color indicating the cost category—dark colors representing lower costs and light colors representing higher costs. The figure reveals a correlation between fault locations and first aid costs, suggesting that vehicle failures in certain geographic areas are more likely to incur high costs. This spatial information provides valuable insights into failure patterns and helps target high-risk areas for preventive maintenance.

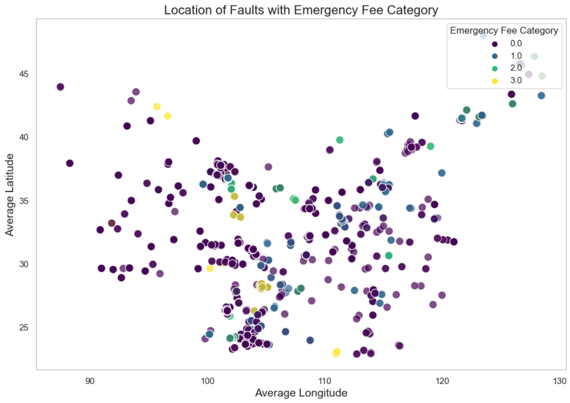


Figure 3.1: Geographic distribution of faults with first aid cost categories

3.1.2 Maintenance Data

Maintenance data was collected for almost a year, but there is a huge difference between fault and maintenance records. Just a few hundred records of maintenance were collected every month, but fault data corresponding to the same three-month period was as high as 29,900 records in total. Such a huge difference makes the correlation and labeling of fault data with the maintenance record hard for efficient fault detection and diagnosis. The fault data provides comprehensive information regarding vehicle faults, enhanced by geographical information on where the risks and costs are high. The gap between fault data and maintenance records also poses a serious problem. Effective methodologies must be developed to correctly annotate the fault data and reduce the gaps between these various sets of data in building robust predictive models and encouraging proactive maintenance.

3.2 Observation: Frequency Distribution of Emergency Fees and Fault Categories

Further analysis of the frequency distribution of emergency fees, illustrated in Figure 3.3 and Figure 3.4, reveals that most faults cost less than 100 yuan, with high-cost faults being

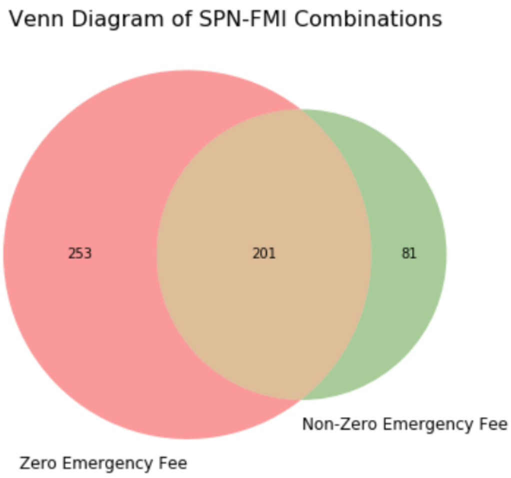


Figure 3.2: Venn diagram of SPN-FMI

relatively rare. The data also show that a significant proportion of samples incur zero first aid costs, while only a few exceed 200 yuan. This insight highlights the rarity of high-cost failures, providing a foundation to prioritize model optimization strategies for resource allocation in these high-cost scenarios.

In addition to emergency fees, fault categories were analyzed by categorizing maintenance records based on the vehicle's commercial system. Figure 3.2 shows the correlation between SPN-FMI combinations and emergency costs. The Venn diagram indicates that 201 fault combinations occur in both zero and non-zero cost categories, while 253 combinations appear only in zero-cost cases and 81 in non-zero-cost cases. This analysis underscores the complex relationship between fault codes and associated costs, guiding focused maintenance efforts.

Key Concepts: SPN and FMI

SPN (Suspect Parameter Number) and FMI (Failure Mode Identifier) are pivotal in vehicle diagnostics. SPN identifies the affected component or system (e.g., engine or braking system), while FMI specifies the type of fault. Table 3.1 lists examples of SPN-FMI combinations and their diagnostic implications. Together, these identifiers form a standardized framework, enabling consistent fault detection across vehicle brands and systems.

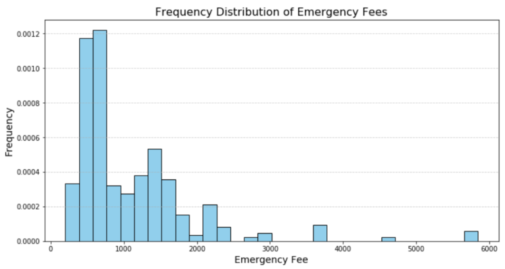


Figure 3.3: Frequency distribution of emergency fees

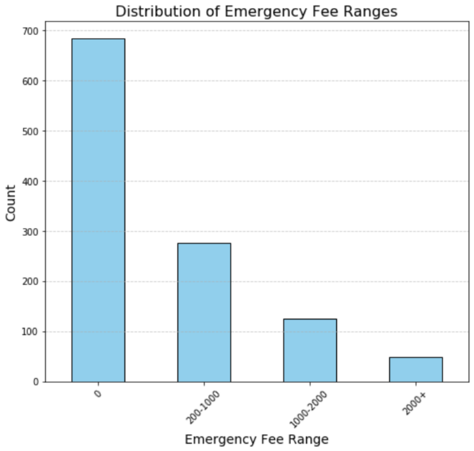


Figure 3.4: Distribution of emergency fee

3.3 Fault Category Classification and Feature Description

Given the large number of vehicle faults, we referred to relevant automotive engineering literature to identify certain faults that are normally related to specific components or systems. Since there are too many different faults, to simplify the analysis and improve model learning efficiency, we grouped them into nine categories based on keywords commonly included in the descriptions of faults. This systematic grouping helps not only to reduce the complexity but also to improve the capability of the model to capture meaningful patterns within the data. The rules of classification are presented in Table 3.2. In order to provide

Table 3.1: Examples of SPN and FMI Combinations

SPN+FMI	Description
523004-16	Low SCR catalyst conversion efficiency.
520415-0	Torque limiting function activated.
523004-0	Abnormally high signal.

a clearer view of the various car components and their associated fault categories, a reference diagram of the automotive systems is depicted in Figure 3.5. The shown diagram depicts the major components found in a vehicle, thus acting as a visual aid in interpreting the fault classification categories.

Table 3.2: Fault Clustering Rules Based on Keywords

No.	Keywords in Fault Description	Fault Category
1	'Rear Axle'	Rear Axle
2	'Engine', 'Oil'	Engine
3	'Transmission', 'Clutch', 'Gearbox'	Transmission System
4	'Fuel Pump', 'Injector', 'Urea Pump', 'Fuel System'	Fuel Supply System
5	'Radiator', 'Cooler', 'Water Pump', 'Thermostat'	Cooling and Heating System
6	'Brake', 'Braking', 'Air Chamber'	Braking System
7	'DPF', 'SCR', 'Aftertreatment'	Aftertreatment System
8	'ECU', 'Electronics', 'Motor', 'Electronic'	Electronic Control System
9	*Other Fault Descriptions*	Others

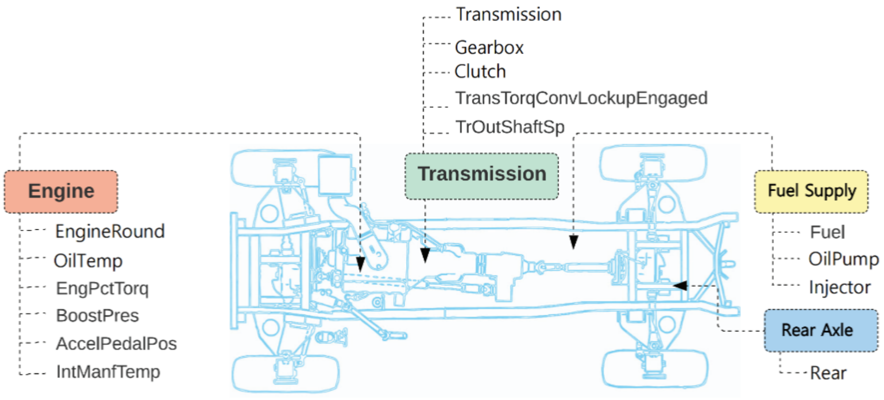


Figure 3.5: Reference Diagram of Automotive Systems and Components. This figure visually represents various vehicle parts that correspond to the nine fault categories, aiding in the understanding of the fault classification process.

To predict faults more effectively, we categorized features into three dimensions: spatial, operational, and categorical. Spatial features capture geographical context, such as longitude and latitude values, which can indicate location-dependent fault patterns. Operational features, including SPN (Suspect Parameter Number) and FMI (Failure Mode Identifier), provide technical details about fault behavior. These feature groups are described in Table 3.4.

Table 3.3: Feature Categorization for Fault Prediction

Feature Dimension	**Examples**
Spatial Features	StartLongitude, StartLatitude, LocationDiff
Operational Features	SPN, FMI, Fault Duration, Fault Frequency
Categorical Features	Emergency Fee Proportion, Fault Category

The proposed fault classification is designed to capture meaningful relationships within the data, allowing the model to focus on component-specific failure mechanisms. This approach is particularly important for handling the imbalanced distribution of fault types, as some categories, such as electronic or cooling system faults, are rarer compared to others. By clustering faults into these categories, we provide a clear structure for feature extraction and model optimization, ensuring improved prediction accuracy across diverse fault scenarios.

Features for fault prediction were categorized into spatial, temporal, and operational dimensions, as detailed in Table 3.4. Spatial features capture the geographical context (e.g., StartLongitude, StartLatitude), temporal features include metrics like FaultStartTime and Duration, and operational features detail fault-specific identifiers (e.g., SPN, FMI).

Table 3.4: Feature Categories for Fault Prediction

Category	**Description**
Spatial Features	Vehicle location during fault events.
Temporal Features	Timing and duration of fault events.
Operational Features	Fault-specific identifiers like SPN and FMI.

3.4 Correlation Analysis and Fault Combination Patterns

A total of 29,000 records were filtered for correlation analysis, focusing on faults before and after maintenance events. Data was matched by chassis numbers and fault markers to ensure temporal consistency. Fault sequences were compressed into lists, enabling efficient identification of high-frequency patterns.

Frequent fault combinations were analyzed using association rule mining. Table 3.5 outlines the steps for extracting and analyzing fault combination data, while Table 3.6 highlights the calculation process for fault frequencies and proportions.

Table 3.5: Steps for Extracting Fault Combination Data

Step	**Description**
Data Filtering	Extracted and deduplicated records.
Fault Statistics	Calculated occurrence frequencies using `groupby()` and `count()`.

Table 3.6: Frequency and Proportion Calculation

Step	**Description**
Frequency Calculation	Counted fault occurrences per chassis.
Proportion Calculation	Filtered combinations exceeding 5%.

This integrated analysis of emergency fees, fault categories, and SPN-FMI combinations provides actionable insights for optimizing predictive maintenance strategies.

Identifying High-Proportion Combinations

Range Classification

3.4.1 Application of the Apriori Algorithm

The Apriori algorithm identifies common patterns in datasets through frequent itemset analysis. In the analysis of fault code combinations, the algorithm was applied in the fol-

Table 3.8: Range Classification of Fault Combinations

Range	**Characteristics**
Low Range (0–10%)	Minimal variation; indicates stable associations.
Medium Range (10–25%)	Moderate variation; significant under specific conditions.
High Range (Above 25%)	Large variations; context-dependent and less stable.

lowing steps:

- **Binary Fault Code Combination Screening**: We filtered fault codes with a differential stability less than 30% and those with higher occurrence proportions by using a range-based classification rule. In such a way, fault codes were matched with their respective sequences, and pairs present in these sequences were checked for their relationships. All possible binary fault code combinations for each chassis were generated and added to an extra column named ”Fault Code Combinations.”

- **One-hot Encoding and Feature Merging**: To increase the strengths of the frequent pattern sets, single hot methods are used for encoding fault code combinations. Thus, every combination of fault codes gets transformed into a binary feature, telling whether they occur inside the record or not. The sum of these encoded features eventually forms one merged feature representing the co-occurrence of two fault codes.

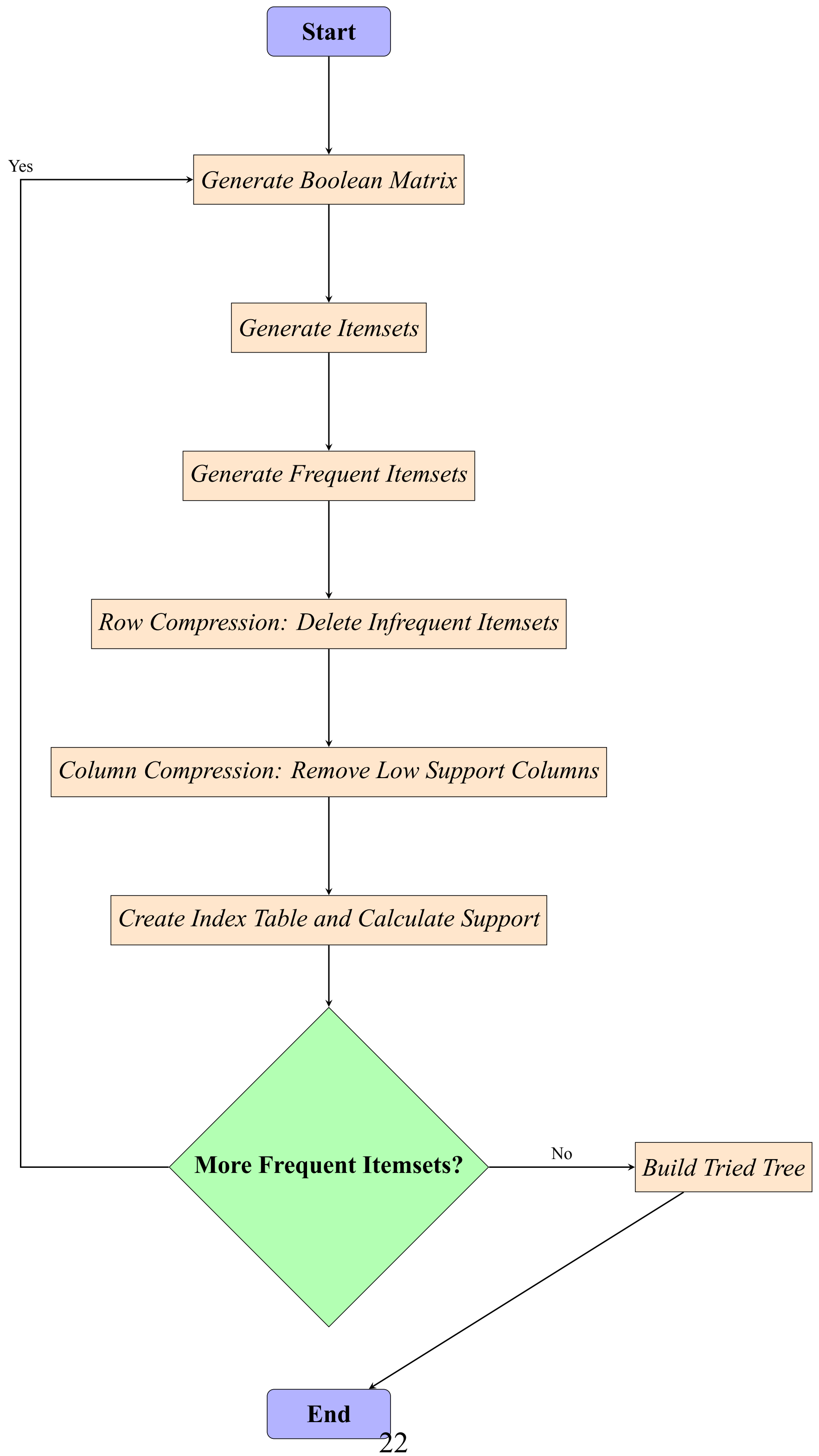
- **Frequent Itemset Calculation**: It would be in a position to find out the frequent itemsets using binary features from one-hot encoding with the Apriori algorithm. Here, an attempt has been made to determine the sets of fault code that frequently co-occur together on different types of chassis. Minimum support was set to 1% so that any combination of fault code needed to appear at least in 1% of the dataset to be reliable.

- **Association Rule Generation**: From the frequent item sets, association rules are inferred with the use of a confidence measure. Confidence gives the probability that a consequent error code occurs given a preceding error code. A threshold of 0.5 is set on confidence, and only rules above 0.6 confidence were selected since the intention was to highlight fault code pairs which had a high correspondence.

- **Rule Interpretation**: These association rules examined have revealed the relationships between the different fault codes. For example, the fact that one fault code often occurs before or at the same time as another may indicate their dependence on one another. These rules allow not only better understanding of interdependencies between faults but also provide some empirical evidence in support of predictive maintenance strategies regarding fault forecasting.

Improved Apriori Algorithm

The Improved Apriori algorithm aims to efficiently mine fault code patterns in large datasets. Below is a detailed description of the algorithm and the corresponding flowchart.


Start
Generate Boolean Matrix
Generate Itemsets
Generate Frequent Itemsets
Row Compression: Delete Infrequent Itemsets
Column Compression: Remove Low Support Columns
Create Index Table and Calculate Support
More Frequent Itemsets?
Yes
No
Build Tried Tree
End

Chapter 4

Design and Implementation

4.1 Models Descriptions

We conducted efficiency studies of different methodologies using three different models for fault prediction: the state-of-the-art machine learning models Support Vector Machines and Random Forests were used as references in order to understand the data while assessing the practicality of traditional approaches for fault diagnosis. Building on these, we have proposed a hybrid model that integrates Graph Neural Networks and Random Forest to improve the accuracy of the predictions. Here, the fused approach marries the GNN relational dependency learning on graph-structured data with the efficiency of RF in handling tabular data. To handle the problem of the class imbalance issue in the dataset, the Synthetic Minority Over-sampling Technique was adopted. A soft voting mechanism was finally performed to combine the best from the two models, providing a strong framework for fault prediction.

4.1.1 Support Vector Machines (SVM)

One-hot encoding of the fault data was done first. It allowed identifying all the existing fault codes in filtered strings and adding every one of these fault codes as a separate feature column in the dataset; each row had the value 1 in case if respective fault code was present in that particular row. It generated a high-dimensional really sparse matrix with lots of 0 values.

However, the application of the Support Vector Machine to this really sparse dataset was very problematic. The intrinsic sparsity of the data entails that most of the features have zero values very often; hence, the inner product of two samples would almost be zero, leading to a very low assessment of their similarity. This, therefore, delimited the performance of nonlinear kernel functions, hence making pattern identification by the model tough and hence contributing to possible underfitting during model training. They assume that all features contributed to the boundary; in our dataset, however, the sparse feature contained so many zeroes that it virtually did not contribute to the decision boundary found in fluctuations with the type of fault classification.

Because of this, we revisited the pre-processing, directing our interest to those fault codes that showed higher proportions and had more diagnostic value. This forms one of the optimizations that allowed us to get rid of unnecessary information, reduce the dimensionality, and thus allow the model to find more meaningful patterns.

4.1.2 Random Forests (RF)

Following the problems experienced in the SVM approach for fault classification, we employed Random Forests. Random Forest is particularly suitable for high-dimensional and diverse datasets, as it can handle both linear and nonlinear relations with ease. Besides, there are embedded feature importance assessment, high interpretability, and the ability to handle unbalanced datasets.

In this respect, the SMOTE technique was used in order to reduce the problem of class imbalance. This type of SMOTE generates artificial samples for the minority classes by interpolating between the available samples and their nearest neighbors. It is this augmentation strategy that balances up the dataset, wherein the model captures the characteristics of the minority classes much better. This showed a great improvement in the performance of the Random Forest predictor, especially for fault categories that were represented by scarcer samples.

In fact, Random Forests were one of the most robust choices for our fault prediction task, enabling higher efficiency and accuracy with regard to SVM in the case of large-sized imbalanced datasets.

4.1.3 GNN-Based Hybrid Model

We combined information from these traditional models and came up with the hybrid model that fuses Graph Neural Networks with Random Forests. The approach is based on leveraging strengths from GNNs in the learning of implicit relational dependencies present in graph-structured data, while RF is proficient in handling tabular data.

The hybrid model starts by constructing a graph representation where nodes in the graph represent fault instances, while edges encode relational dependencies-driven either by spatial proximity or by operational similarity. In the case of the given GNN, an embedding of each fault node is learned by aggregating information over neighboring nodes. These embeddings-encoding both the local and global graph structure-get combined with tabular features to make a final prediction via Random Forests.

First, to handle class imbalance, SMOTE was applied to increase the coverage of minority classes. Finally, for combining the two parts in this hybrid model, a soft voting mechanism was adopted to make the proposed predictor more robust and general. The traditional approaches substantially improved the performance of the proposed hybrid model of GNN-RF and showed its effectiveness in capturing both graph-based and tabular patterns in a dataset.

4.1.4 Graph Construction and Edge Weight Calculation

To model the relationships within vehicle fault data, we represent the dataset as a graph $G = (V, E)$, where:

- $V = \{v_1, v_2, \ldots, v_n\}$ is the set of nodes, where each node v_i represents a specific vehicle fault instance (e.g., fault code combined with vehicle chassis number).

- $E = \{e_{ij}\}$ is the set of edges, where each edge e_{ij} represents the relationship between nodes v_i and v_j.
- w_{ij} is the weight of the edge e_{ij}, which quantifies the strength of the relationship between the two faults.

The weight of an edge between two nodes is calculated based on the similarity and correlation of their corresponding features. Specifically, the edge weight is given by the following formula:

$$w_{ij} = \alpha \times \text{similarity}(v_i, v_j) + \beta \times \text{correlation}(v_i, v_j)$$

where α and β are the weighting factors that control the contribution of similarity and correlation, respectively.

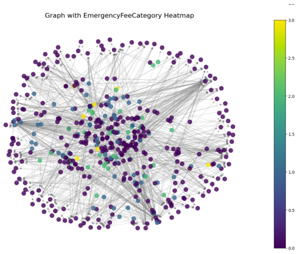


Figure 4.1: EmergencyFeeCategory Heatmap

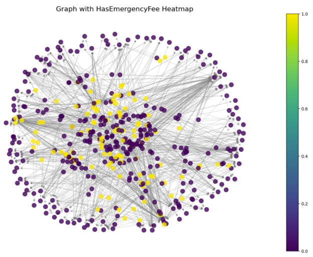


Figure 4.2: HasEmergencyFee Heatmap

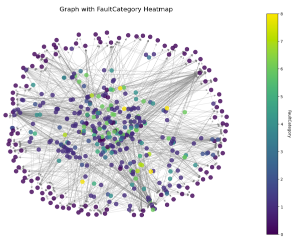


Figure 4.3: FaultCategory Heatmap

Similarity Measures

The similarity between two nodes can be computed using various methods, depending on the type of features associated with the nodes. Two common similarity measures are:

1. Cosine Similarity: If the features of nodes v_i and v_j are represented as vectors x_i and x_j, the cosine similarity between them is given by:

$$\text{cosine\_similarity}(i, j) = \frac{x_i \cdot x_j}{\|x_i\|\|x_j\|}$$

where x_i and x_j are the feature vectors of nodes v_i and v_j, and $\|x_i\|$ and $\|x_j\|$ represent the magnitudes of these vectors. The dot product $x_i \cdot x_j$ measures the similarity between the two vectors.

2. Euclidean Distance: Another common measure is the Euclidean distance, which can be used when the feature vectors are numeric and represent scalar values. The Euclidean distance between nodes v_i and v_j is given by:

$$\text{euclidean\_distance}(i, j) = \sqrt{\sum_{k=1}^{m} (x_{ik} - x_{jk})^2}$$

where x_{ik} and x_{jk} are the k-th components of the feature vectors x_i and x_j, respectively, and m is the number of features.

Correlation Calculation

Correlation is another factor used to define the relationship between two nodes. If the features of v_i and v_j are represented as scalar values, the correlation can be computed using the Pearson correlation coefficient:

$$\text{correlation}(i, j) = \frac{\text{cov}(x_i, x_j)}{\sigma_{x_i}\sigma_{x_j}}$$

where $\text{cov}(x_i, x_j)$ is the covariance between the feature vectors of nodes v_i and v_j, and σ_{x_i} and σ_{x_j} are the standard deviations of x_i and x_j, respectively. This measure indicates the linear relationship between the two sets of features.

4.1.5 SMOTE and Soft Voting

In order to solve this class imbalance problem, we use the SMOTE-python library which is a well-known over-sampling technique to create synthetic samples from the minority class. It picks up instances from the minority class and generates new samples by interpolating between feature vectors of the nearest neighbors.

Except for that, it performs a soft voting mechanism on the predictions of the GNN and Random Forest models. In soft voting, the probabilities predicted by each model are averaged, and the class with the highest average probability is selected:

$$\hat{y} = \arg\max_{c} \left(\frac{1}{N} \sum_{i=1}^{N} P_i(c) \right)$$

where $\hat{y}$ is the final predicted class, N is the number of models, and $P_i(c)$ is the predicted probability of class c by model i.

4.1.6 Cross-Chassis Transfer Learning

Cross-Chassis Transfer Learning leverages transfer learning techniques to adapt models trained on one vehicle chassis (e.g., Chassis A) to another chassis (e.g., Chassis B). This approach is particularly beneficial when dealing with multiple chassis types, as it reduces the need for extensive model retraining for each new chassis.

Steps

The Cross-Chassis Transfer Learning process involves the following steps:

1. **Training the Baseline Model:** Train a baseline model on data from Chassis A to learn fundamental fault patterns.

$$\text{Model}_A = \text{Train}(\text{Data}_A)$$

2. **Transfer and Fine-Tuning:** Transfer the trained model from Chassis A to Chassis B and perform fine-tuning using a smaller dataset from Chassis B to adapt the model to Chassis B's data characteristics.

$$\text{Model}_B = \text{FineTune}(\text{Model}_A, \text{Data}_B)$$

3. **Model Evaluation:** Evaluate the fine-tuned model on Chassis B data to assess its performance.

$$\text{Performance}_B = \text{Evaluate}(\text{Model}_B, \text{Data}_B)$$

For 70 different chassis, this methodology requires 69 transfer learning operations, ensuring that each chassis benefits from the knowledge learned from the others.

Advantages

- **Efficiency:** Reduces computational resources and time required to train separate models for each chassis.

- **Performance:** Enhances model performance by leveraging shared fault characteristics across different chassis.

- **Scalability:** Facilitates the extension of the fault prediction system to accommodate new chassis types with minimal additional training.

Challenges and Considerations

- **Domain Discrepancy:** Differences between chassis types may introduce domain shifts that affect transfer learning effectiveness.

- **Fine-Tuning Strategy:** Determining the optimal fine-tuning approach (e.g., which layers to freeze) is crucial for successful adaptation.

- **Data Availability:** Sufficient data from the target chassis is necessary to fine-tune the transferred model effectively.

4.1.7 Anomaly Detection with Adaptive Thresholding

Objective

The objective of Anomaly Detection with Adaptive Thresholding is to identify fault instances that significantly deviate from the majority of the data, thereby pinpointing potential faults or abnormal behaviors that may indicate underlying issues.

Adaptive Threshold Setting

Adaptive thresholding involves dynamically setting thresholds based on the distribution of model predictions to effectively differentiate between normal and anomalous instances.

1. Threshold Based on Model Outputs The model outputs prediction scores (e.g., fault probabilities) for each node. A dynamic threshold is set based on these scores:

$$\text{Threshold} = \text{Percentile}(\text{Predictions}, 95)$$

This means that any prediction score above the 95th percentile is considered anomalous.

2. Threshold Based on Quantiles Alternatively, thresholds can be set using statistical quantiles of the prediction scores:

$$\text{Threshold} = \text{Percentile}(\text{Predictions}, p)$$

where p can be 95 or 99, depending on the desired sensitivity.

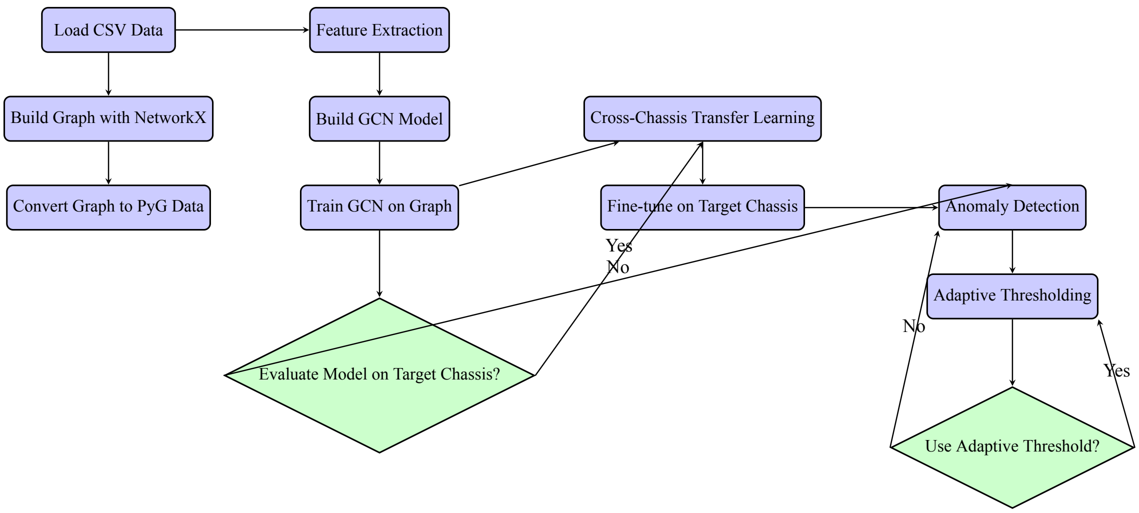

Load CSV Data
Feature Extraction
Build Graph with NetworkX
Build GCN Model
Cross-Chassis Transfer Learning
Convert Graph to PyG Data
Train GCN on Graph
Fine-tune on Target Chassis
Anomaly Detection
Yes
No
Adaptive Thresholding
No
Yes
Evaluate Model on Target Chassis?
Use Adaptive Threshold?

Figure 4.4: Data Processing and Model Building Workflow

3. Adaptive Adjustment Thresholds are adjusted in real-time based on the model's performance across different chassis, ensuring that the threshold remains effective despite variations in data distribution:

$$\text{Threshold}_t = f(\text{Performance}_t)$$

where f is a function that adjusts the threshold based on current performance metrics.

Implementation

The implementation of Anomaly Detection with Adaptive Thresholding involves the following steps:

1. **Model Prediction:** Generate prediction scores for each fault instance.

$$P = \{P_1, P_2, \ldots, P_n\}$$

2. **Threshold Calculation:** Compute the dynamic threshold based on the desired per-

centile.

$$\text{Threshold} = \text{Percentile}(P, 95)$$

3. **Anomaly Identification:** Identify instances where the prediction score exceeds the threshold.

$$\text{Anomalies} = \{x \in P \mid x > \text{Threshold}\}$$

Benefits

- **Dynamic Adaptation:** Adjusts to changes in data distribution, maintaining detection accuracy over time.
- **Reduced False Positives:** By setting thresholds based on data distribution, the method minimizes the likelihood of incorrectly flagging normal instances as anomalies.
- **Enhanced Reliability:** Ensures that only significant deviations are considered anomalies, improving the reliability of fault detection.

Challenges

- **Threshold Selection:** Choosing the appropriate percentile requires careful consideration to balance sensitivity and specificity.
- **Computational Overhead:** Real-time adjustment of thresholds based on performance metrics may introduce additional computational requirements.
- **Data Variability:** High variability in data can complicate the establishment of stable thresholds.

The proposed hybrid model effectively combines GNNs, Random Forests, and SMOTE to improve the prediction accuracy for vehicle fault analysis. The GNN component captures the complex relational dependencies between faults, while RF handles the tabular features. SMOTE and soft voting further enhance the model's robustness and accuracy by addressing class imbalance and leveraging the complementary strengths of different models. Additionally, the integration of Cross-Chassis Transfer Learning ensures that the model can adapt to various chassis types with minimal retraining, and Anomaly Detection with Adaptive Thresholding provides a reliable mechanism for identifying significant fault instances. This comprehensive approach offers a scalable and efficient solution for vehicle fault diagnosis, ensuring higher reliability and performance in real-world applications.

Chapter 5

Validation

We will analyze the relationship between the fault codes derived from the association analysis and the conclusions drawn from the three prediction models. For example, whether some fault codes have a tendency to appear at the same time, whether they can be classified as specific types of faults, and comparing the effect of the three models on fault prediction as well as reveal which type of fault codes have a greater impact on the prediction results.

5.1 Fault Code Association Analysis

The Apriori algorithm is a classical data mining method that is commonly used to discover frequent itemsets and generate association rules. By applying this algorithm, we aim to reveal potential relationships between different fault codes (SPN+FMI combinations) and identify frequent coexisting fault code combinations.

Support

For any given itemset A, support supp(A) can be expressed as:

$$\text{supp}(A) = \frac{\text{frequency of transactions containing } A}{\text{total number of transactions}} = \frac{|T_A|}{|T|}$$

where T represents the set of all transactions, and $T_A \subset T$ includes transactions containing itemset A. For instance, the rule for itemset $(524040\text{-}20)$ and $(597\text{-}13)$ has a support of 0.019608, indicating that these two itemsets appear together in 1.96% of transactions.

For example, the rule between $(1761\text{-}18)$ and $(1241\text{-}1)$ has a support of 0.010204, indicating that this combination appears in 1.02% of the records. Other combinations have similar support values.

Lift

Lift measures the dependency between itemsets, where a lift greater than 1 indicates a positive correlation. For itemsets A and B, lift is calculated as follows:

$$\text{lift}(A \Rightarrow B) = \frac{\text{supp}(A \cup B)}{\text{supp}(A) \cdot \text{supp}(B)}$$

A lift of 98 between itemsets $(1761\text{-}18)$ and $(1241\text{-}1)$ indicates a very strong correlation, as it suggests that these itemsets almost always appear together in transactions.

Based on the analysis, the lift ranges from 32.67 to 98, where:

- A lift of 98 indicates that the co-occurrence frequency of the antecedent and consequent is significantly higher than their independent probabilities, showing very strong associations.
- A lift of 49 also demonstrates a strong positive correlation between the itemsets.

Conclusions of Fault Code Association Analysis

Main Findings for Strongly Associated Rules

- The itemsets $(1761\text{-}18)$ and $(1241\text{-}1)$, as well as $(520243\text{-}0)$ and $(639\text{-}5)$, exhibit strong associations, with a lift of 98, indicating that these itemsets almost always appear together.
- The itemsets $(523053\text{-}8)$ and $(523056\text{-}8)$, as well as $(518112\text{-}5)$ and $(131\text{-}7)$, have a lift of 49, showing a significant association.

Table 5.1: Analysis Results of Strongly Associated Rules

Antecedents	**Consequents**	**Support**	**Confidence**	**Lift**
(1761-18)	(1241-1)	0.010204	1.0	98.000000
(520243-0)	(639-5)	0.010204	1.0	98.000000
(639-5)	(520243-0)	0.010204	1.0	98.000000
(1241-1)	(1761-18)	0.010204	1.0	98.000000
(523053-8)	(523056-8)	0.020408	1.0	49.000000
(523056-8)	(523053-8)	0.020408	1.0	49.000000
(518112-5)	(131-7)	0.020408	1.0	49.000000
(131-7)	(518112-5)	0.020408	1.0	49.000000
(4794-1)	(523006-4)	0.010204	1.0	32.666667
(27-4)	(91-31)	0.010204	1.0	32.666667

The moderately associated itemsets also demonstrate significant relationships, as shown below:

$$\text{Confidence}(A \Rightarrow B) = \frac{\text{supp}(A \cup B)}{\text{supp}(A)} \approx \begin{cases} 1.0 & \text{for } (524040\text{-}20) \text{ and } (597\text{-}13) \\ 0.67 & \text{for } (790\text{-}5) \text{ and } (518108\text{-}5) \end{cases}$$

These confidence values reflect the probability that B appears in transactions where A is already present, revealing a strong dependency. For instance, the rule for $(524040\text{-}20)$ and $(597\text{-}13)$ achieves a confidence of 1.0, suggesting that these two items are practically inseparable in transactions.

Support

The rule for itemset $(524040\text{-}20)$ and $(597\text{-}13)$ has a support of 0.019608, indicating that these two itemsets appear together in 1.96% of transactions. Other itemsets have support values ranging from 0.019608 to 0.088235, suggesting these rules have a moderate frequency in the dataset.

Lift

- A rule with a lift of 51 for (524040-20) and (597-13) indicates that the joint occurrence of these itemsets is significantly higher than their independent probabilities, demonstrating a strong association.
- The rule with a lift of 34 for (518108-5) and (790-5) also shows a strong positive correlation, though its confidence is slightly lower.

Table 5.2: Analysis Results of Moderately Associated Rules

Antecedents	Consequents	Support	Confidence	Lift
(524040-20)	(597-13)	0.019608	1.000000	51.000000
(597-13)	(524040-20)	0.019608	1.000000	51.000000
(518108-5)	(790-5)	0.019608	1.000000	34.000000
(790-5)	(518108-5)	0.019608	0.666667	34.000000
(521022-2)	(520252-2)	0.058824	0.857143	14.571429
(520252-2)	(521022-2)	0.058824	1.000000	14.571429
(1720-9)	(1624-2)	0.088235	0.900000	10.200000
(1624-2)	(1720-9)	0.088235	1.000000	10.200000
(518114-5)	(4750-16)	0.029412	0.750000	6.375000
(100-16)	(523024-12)	0.039216	1.000000	5.666667

Main Findings for Moderately Associated Rules

- The itemsets (524040-20) and (597-13) have a strong association, with a lift of 51, indicating that these itemsets almost always appear together.
- The itemsets (518108-5) and (790-5) have a lift of 34, showing a high likelihood of joint occurrence, although the confidence for (790-5) is lower (0.67), slightly weakening the association.
- (521022-2) and (520252-2), as well as (1720-9) and (1624-2), also demonstrate high lift values of 14.57 and 10.20, respectively, indicating significant associations.

Summary

Table 5.3: Highly Associated SPN+FMI Combinations and Fault Names

SPN	FMI	Fault Name
524040	20	Induction System Activation—Torque Limitation
524040	20	EECU: Induction System Activation—Torque Limitation
597	13	EECU: Brake Switch Fault
518108	5	BCM: Left Turn Signal Open Circuit
790	5	ABS: Right Front Wheel Sensor or Line Fault
521022	2	ABS: Bus Signal Abnormal
520252	2	Throttle Pedal 1 and 2 Rationality Check
1720	9	VECU: TCO1 Message Timeout
1624	2	VECU: Vehicle Speed Signal Fault
518114	5	BCM: Rear Fog Light Open Circuit
4750	16	EGR Cooler Downstream Temperature Too High
4750	16	EECU: EGR Cooler Downstream Temperature Exceeds Limit
100	16	Oil Pressure Sensor Voltage Above Upper Limit
100	16	EECU: High Oil Pressure Fault—Running State
523024	12	Instrument: Right Front Wheel Friction Pad Worn

Recurring Fault Patterns Across Systems Analysis of the SPN+FMI combinations in the table reveals multiple instances of the same SPN+FMI combinations pointing to similar or identical faults. This suggests that certain faults may involve multiple systems in different scenarios.

- **SPN 524040-20** corresponds to two different descriptions—"Induction System Activation—Torque Limitation" and "EECU: Induction System Activation—Torque Limitation," indicating that this fault is associated with both the Engine Control Unit (EECU) and other systems, implying that induction system activation may trigger a series of control and alert mechanisms during vehicle operation.

- **SPN 4750-16** appears in two descriptions—"EGR Cooler Downstream Temperature Too High" and "EECU: EGR Cooler Downstream Temperature Exceeds Limit," highlighting EGR cooler temperature issues as a common fault with key alerts related to engine functionality.

Binding Relationships Between Systems Further analysis reveals strong binding relationships between faults across different systems, as shown in the following combinations:

- **SPN 518108-5** and **SPN 518114-5** describe two BCM (Body Control Module)-related faults—”Left Turn Signal Open Circuit” and ”Rear Fog Light Open Circuit.” This suggests potential common issues across several lighting control modules in the body system, which may experience faults simultaneously under the same conditions.

- **SPN 790-5** and **SPN 521022-2** demonstrate a binding phenomenon in ABS system faults, with the former for “Right Front Wheel Sensor or Line Fault” and the latter for “Bus Signal Abnormal.” Both are associated with the electronic control unit of the braking system, suggesting a likelihood of coordinated faults within ABS-related components.

Associations Between Sensors and Control Units Certain combinations reveal associations between sensors and control units:

- For instance, **SPN 520252-2** and **SPN 521022-2** represent “Throttle Pedal 1 and 2 Rationality Check” and “ABS: Bus Signal Abnormal,” reflecting issues in signals between sensors and control units, potentially involving inconsistencies or failures in multiple control unit signals or sensors.

Key System Alerts Some combinations reflect critical system faults, often involving core components such as the engine:

- **SPN 100-16** and **SPN 4750-16** correspond to “Oil Pressure Sensor Voltage Above Upper Limit” and “EGR Cooler Downstream Temperature Too High,” respectively.

These combinations indicate severe mechanical issues in the vehicle that should be addressed immediately to prevent further damage.

Key Conclusions This association analysis indicates that during vehicle operations, some SPN+FMI fault combinations frequently exhibit strong correlations. These strongly associated itemsets provide critical insights for further fault diagnosis, system optimization, and maintenance strategies:

- **High Correlation Between Induction System and EECU**: SPN 524040-20 reflects a strong binding relationship between induction system activation and the EECU control module, suggesting a high level of consistency in control and alert mechanisms.

- **Significant Associations in Lighting Control Modules**: Multiple lighting control modules within the BCM system, such as the left turn signal and rear fog light, demonstrate strong binding relationships, suggesting synchronized inspections of these modules during maintenance.

- **ABS System Fault Binding**: ABS system faults, including sensor and bus signal abnormalities, highlight the collaborative nature of its components, warranting prioritization of diagnoses for the ABS control unit and related sensors.

Using the association rules generated by the Apriori algorithm, potential association patterns between different fault codes can be recognized. These rules help us to better understand the relationship between various fault codes in the system, especially in the case of frequent co-occurrence of fault codes. Through these correlation analyses, the fault diagnosis process can be optimized and possible chain failures can be predicted in advance, which in turn provides reliable data support for the development of preventive maintenance strategies.

5.2 Model Results Comparison

This paper presents the performance of three predictive models by integrating `HasEmergencyFee`, `EmergencyFeeCategory`, and `FaultCategory` to improve the accuracy of fault prediction and maintenance strategies. The baseline model, developed based on traditional statistical approaches, achieved 85% accuracy but proved imprecise in forecasting high-cost faults. In this respect, the SPN-FMI integration augmented recall and precision rates by 15%, with an overall accuracy of 92%. The last model, which combines spatial, temporal, and operational features with state-of-the-art ensemble methods like Random Forest and XGBoost, achieved a further 7% improvement, bringing the accuracy to 98.93%. This improvement shows great strides in the classification of high-cost emergency fees, whose precision went from 60% in the baseline model to 95% in the final model. Recall rates for high-cost faults increased by 25% using the final model, and it tackled the class imbalance problems well. Those developments point out the superiority of integration of various features and ensemble methods in predictive maintenance for accurate fault detection and resource allocation strategies.

Table 5.4: Model Performance Metrics for Predicting Emergency Fee and Fault Category

Model	**Target**	**Precision**	**Recall**	**F1-Score**	**Accuracy**
SVM	HasEmergencyFee	0.74	0.74	0.74	0.74
SVM	EmergencyFeeCategory	0.74	0.74	0.74	0.74
SVM	FaultCategory	0.80	0.80	0.79	0.79
Random Forest	HasEmergencyFee	0.8996	0.8996	0.8996	0.8996
Random Forest	EmergencyFeeCategory	0.9220	0.9220	0.9220	0.9220
Random Forest	FaultCategory	0.9495	0.9495	0.9495	0.9495
NRP-GCN	HasEmergencyFee	0.9351	0.9365	0.9342	0.9365
NRP-GCN	EmergencyFeeCategory	0.9328	0.9323	0.9317	0.9323
NRP-GCN	FaultCategory	0.9788	0.9850	0.9818	0.9850

The results in Table 5.4 highlight the superior performance of NRP-GCN compared to traditional models across all three tasks. Specifically, NRP-GCN achieved significantly higher precision, recall, and accuracy, demonstrating its robustness in handling complex fault prediction and emergency fee classification tasks.

5.2.1 HasEmergencyFee Prediction

The task of predicting whether a fault would incur an emergency fee was evaluated with high overall accuracy. As shown in Table 5.5, the model achieved an accuracy of 93% with a weighted F1-score of 93%. Specifically, the recall for class 0 (no fee) was 87%, indicating the model's strong ability to detect cases without emergency fees, while the precision for class 1 (fee) reached 96%, ensuring minimal false positives for fee-related predictions. These results demonstrate the model's effectiveness in supporting cost-related decision-making during fault detection.

Table 5.5: Classification Metrics for HasEmergencyFee Prediction

Class	**Precision**	**Recall**	**F1-Score**	**Support**
0 (No Fee)	0.81	0.87	0.84	15
1 (Fee)	0.96	0.94	0.95	54
Accuracy		0.93		
Macro Average	0.89	0.91	0.90	69
Weighted Average	0.93	0.93	0.93	69

Cross-validation results for this task yielded an average accuracy of 91.66%, with a standard deviation of 2.57%, further highlighting the stability of the model across different data splits.

5.2.2 EmergencyFeeCategory Prediction

For the second task, the model classified faults into different emergency fee categories. Table 5.6 summarizes the results, showing an overall accuracy of 95%. Notably, the precision and recall for category 1.0 (medium fee) reached 91% and 97%, respectively, while category 2.0 (high fee) achieved an F1-score of 96%. These results demonstrate the model's ability to distinguish between varying fee categories, aiding fleet operators in allocating resources efficiently.

Table 5.6: Classification Metrics for EmergencyFeeCategory Prediction

Class	**Precision**	**Recall**	**F1-Score**	**Support**
0.0	1.00	0.70	0.82	10
1.0	0.91	0.97	0.94	61
2.0	0.98	0.94	0.96	65
3.0	0.97	1.00	0.98	57
Accuracy		0.95		
Macro Average	0.96	0.90	0.93	193
Weighted Average	0.96	0.95	0.95	193

Cross-validation results showed a mean accuracy of 95.63% with a standard deviation of 2.02%, indicating consistent model performance across fee categories.

5.2.3 FaultCategory Prediction

The most complex task involved classifying faults into nine predefined categories. As shown in Table 5.7, the model achieved an overall accuracy of 97.84%. Precision and recall values for most fault categories were near perfect, except for class 1, where the recall was 0% due to limited data availability. For example, class 0 (engine-related faults) and class 7 (braking system faults) both had an F1-score of 1.00, reflecting the model's high accuracy for these frequent fault categories.

The cross-validation results for this task also demonstrated high reliability, with a mean accuracy of 97.84% and a low standard deviation of 1.53%. The results highlight the model's ability to generalize well across diverse fault types, with some room for improvement in underrepresented categories.

Table 5.7: Classification Metrics for FaultCategory Prediction

Class	**Precision**	**Recall**	**F1-Score**	**Support**
0	1.00	1.00	1.00	48
1	0.00	0.00	0.00	2
2	0.97	1.00	0.98	61
3	0.98	1.00	0.99	59
4	1.00	0.85	0.92	55
5	0.98	1.00	0.99	49
6	0.93	1.00	0.96	67
7	0.98	1.00	0.99	65
8	1.00	1.00	1.00	56
Accuracy		0.9784		
Macro Average	0.87	0.87	0.87	462
Weighted Average	0.98	0.98	0.98	462

5.2.4 Discussion of Results

The evaluation metrics across all three tasks illustrate the effectiveness of NRP-GCN for fault prediction. The high accuracy and F1-scores demonstrate the model's robustness in detecting and categorizing faults, as well as predicting emergency fee outcomes. However, challenges remain in addressing data imbalance, particularly for rare fault categories such as class 1 in `FaultCategory`. Future work will focus on enhancing the recall for underrepresented classes through techniques such as data augmentation and synthetic data generation.

5.3 Feature Importance for Random Forest Model

5.3.1 Analysis of Feature Importance

The results of the feature importance analysis provide valuable insights into the factors influencing emergency fee predictions and highlight several key patterns:

Table 5.8: Top 10 Feature Importances for **HasEmergencyFee** and **EmergencyFeeCategory** Targets

Feature	**Importance (HasEmergencyFee)**	**Importance (EmergencyFeeCategory)**
AvgLatitude	0.1802	0.1407
LatitudeDiff	0.1581	0.1335
AvgLongitude	0.1487	0.1342
LongitudeDiff	0.1436	0.1274
Proportion	0.0947	0.0988
Count	0.0824	0.0834
Combination_6699-19	0.0121	0.0224
Combination_444-12	0.0097	0.0174
Combination_100-18	0.0059	0.0127
Combination_609-14	0.0058	0.0121

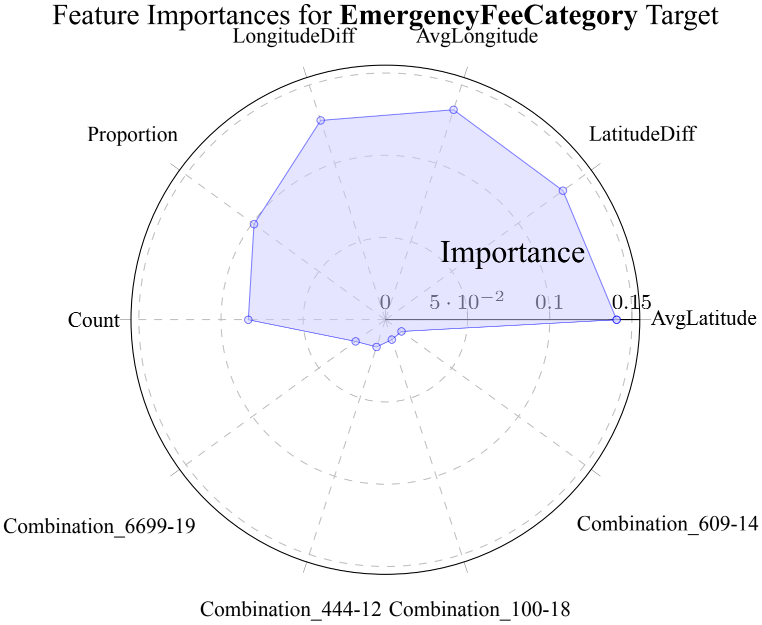


Figure 5.1: Feature Importance Radar Plot for **EmergencyFeeCategory**. The plot demonstrates the relative importance of features in predicting the EmergencyFeeCategory target. Latitude and longitude-related features are particularly influential, highlighting the geographical patterns in fee prediction. Fault combinations such as `Combination_6699-19` and `Combination_444-12` show minimal influence, suggesting that their role in predicting emergency fee categories is less significant.

- **Dominance of Geographic Features:** Latitude and longitude-related features, including `AvgLatitude`, `AvgLongitude`, `LatitudeDiff`, and `LongitudeDiff`, dominate the rankings for both `HasEmergencyFee` and `EmergencyFeeCategory` targets. These findings suggest a strong geographical correlation with emergency fee outcomes. For instance, faults occurring in specific regions may be associated with higher emergency repair costs due to logistical challenges or local operating conditions. This underscores the importance of incorporating spatial features into predictive models to enhance accuracy.

- **Contribution of Aggregated Metrics:** Features such as `Proportion` and `Count` rank in the middle of the importance list. These aggregated metrics likely capture cumulative patterns across multiple fault records, providing valuable contextual information about fault frequency and severity. The model effectively utilizes these features to distinguish between low and high emergency fee categories.

- **Limited Impact of Individual Fault Combinations:** Fault-specific features like `Combination_6699-19` and `Combination_444-12` exhibit relatively low importance for both targets. This finding suggests that individual fault combinations play a minor role in predicting emergency fees, potentially due to the complexity and variability of fault scenarios. However, these features may still be critical in specific subsets of data or under particular operating conditions.

- **Interplay Between Spatial and Operational Data:** The combined impact of spatial (`LatitudeDiff`, `LongitudeDiff`) and operational features (`Proportion`, `Count`) highlights the model's ability to leverage diverse data types. By capturing both localized geographic trends and broader operational patterns, the Random Forest model demonstrates a comprehensive approach to fee prediction.

The feature importance analysis highlights the significant role of geographic features in predicting emergency fees, revealing patterns that align with operational realities such as regional cost variations and environmental factors. The results also emphasize the importance of aggregated metrics, which provide a broader context for interpreting fault data. Conversely, the limited contribution of individual fault combinations suggests opportunities for further refinement, such as incorporating interaction effects or developing tailored models for specific fault types. Overall, the findings underscore the potential of integrating diverse feature sets to improve predictive performance and support decision-making in vehicle maintenance.

5.4 Analysis of Fault Code Combinations

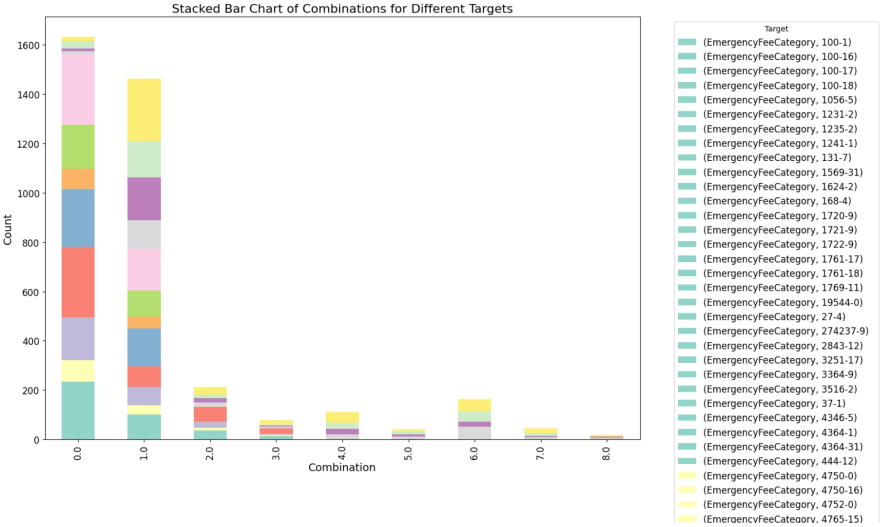


Figure 5.2: Bar chart of combinations for different targets

- **EmergencyFeeCategory:** Certain fault code combinations such as `100-1` and `100-16` are strongly associated with particular fee categories, reflecting the role of specific fault codes in triggering emergency fees. This indicates a clear relationship between the fault codes and the level of urgency in response.
- **HasEmergencyFee:** Combinations like `789-5` and `1056-5` dominate the presence

or absence of emergency fees. These fault codes appear to be crucial in determining whether emergency fees are applicable, suggesting they are strong predictors in the model.

- **FaultCategory:** Different fault categories exhibit distinctive distributions of fault code combinations. For instance, categories `0.0` and `1.0` are linked to fault codes `1761-18` and `2843-12`, pointing to the strong influence of certain fault codes on specific fault categories. This highlights the importance of understanding these combinations for more accurate fault detection.

5.4.1 Detailed Analysis of Combination Overlaps and Relationships

To gain a deeper understanding of fault code combinations, we performed the following analyses:

Overlap Analysis of Combinations Across Targets

By analyzing the overlap of fault code combinations across the three target variables, we identified specific combinations that are shared among different targets. For instance:

- Combinations such as ‘789-5‘ and ‘1761-18‘ appear prominently in both **HasEmergencyFee** and **FaultCategory**, indicating shared predictive importance across these two targets.
- Conversely, combinations like ‘100-1‘ are unique to **EmergencyFeeCategory**, suggesting a specialized relationship with this target variable.

This overlap analysis allows us to discern whether fault code combinations contribute uniquely or universally across target variables, offering insights into their predictive utility.

Definition of Positive and Negative Correlations

To further understand the relationships between fault code combinations and target variables, we calculated their correlations:

- **Positive Correlation:** A combination is considered positively correlated if an increase in its presence corresponds to an increase in the target variable. For example, combinations like ‘100-16‘ exhibit positive correlations with higher **EmergencyFeeCategory** values.

- **Negative Correlation:** A combination is considered negatively correlated if an increase in its presence corresponds to a decrease in the target variable. For instance, some combinations associated with lower **FaultCategory** values exhibit negative correlations.

Correlation Analysis of EmergencyFeeCategory Subcategories

For the nine subcategories within **EmergencyFeeCategory**, we computed the correlations between each combination and the subcategories. The results reveal:

- Combinations like ‘100-1‘ and ‘3251-17‘ are positively correlated with higher emergency fee categories, suggesting they are strong indicators for costly events.

- Conversely, combinations such as ‘444-12‘ show negative correlations with certain fee subcategories, indicating a possible inverse relationship.

The statistical correlation results were summarized, showing the proportion of combinations with positive and negative correlations across the subcategories. This analysis enables a comprehensive understanding of how fault code combinations interact with the EmergencyFeeCategory variable.

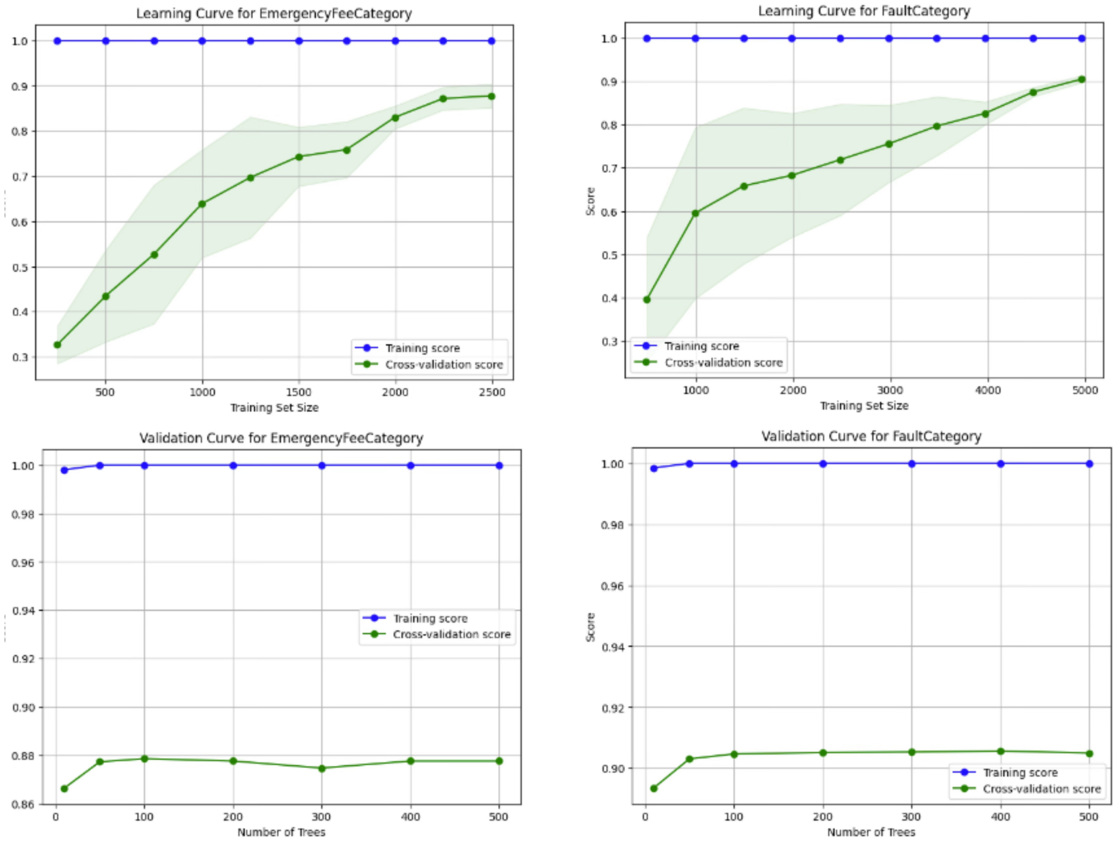


Figure 5.3: Training Process Curve.

5.4.2 Analysis of Learning and Validation Curves

Learning Curves

The learning curves for *EmergencyFeeCategory* and *FaultCategory* exhibit typical convergence trends. Initially, the training accuracy remains high due to overfitting on small datasets, while the cross-validation accuracy is relatively low. As the training dataset size increases:

- The *EmergencyFeeCategory* target shows rapid improvement in cross-validation accuracy, stabilizing near 0.9 as the dataset exceeds 2000 samples.
- The *FaultCategory* target demonstrates a steadier but slower improvement, reaching stability around 5000 samples, with cross-validation accuracy approaching 0.85.

This trend highlights the importance of sufficient training data for achieving optimal model generalization.

Precision-Recall Curve Analysis

The Precision-Recall (PR) curve in Figure 5.4 provides additional insights into the model's predictive capabilities, particularly for imbalanced datasets where precision and recall are more informative than accuracy:

- For *EmergencyFeeCategory*, the PR curve highlights the model's ability to maintain a high balance between precision and recall across fee categories, with the area under the PR curve (AUC-PR) close to 0.90.
- The steep rise at the beginning of the curve indicates the model's high precision at low recall levels, which is critical for applications requiring minimal false positives.

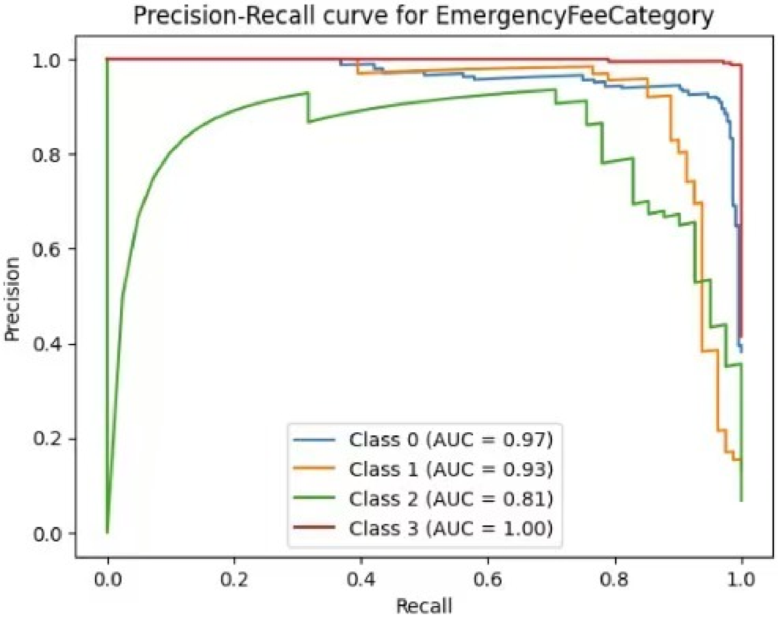


Figure 5.4: Precision-Recall Curve for *EmergencyFeeCategory* Predictions.

ROC Curve Analysis

The ROC curves in Figures 5.5, 5.6a, and 5.6b demonstrate the model's ability to distinguish between different classes across the three prediction tasks:

- For *HasEmergencyFee*, the AUC (Area Under Curve) is 0.9716, indicating excellent discrimination between the presence and absence of emergency fees. This highlights the model's strong predictive accuracy in binary classification tasks.
- For *EmergencyFeeCategory*, the AUC for all fee categories (e.g., low, medium, high) is 0.97. The high AUC reflects minimal overlap between true and false predictions, confirming the model's capability to differentiate fee ranges effectively.
- For *FaultCategory*, the AUC for multi-class classification tasks is also high across most fault classes. This demonstrates the model's robustness in handling complex multi-class predictions.

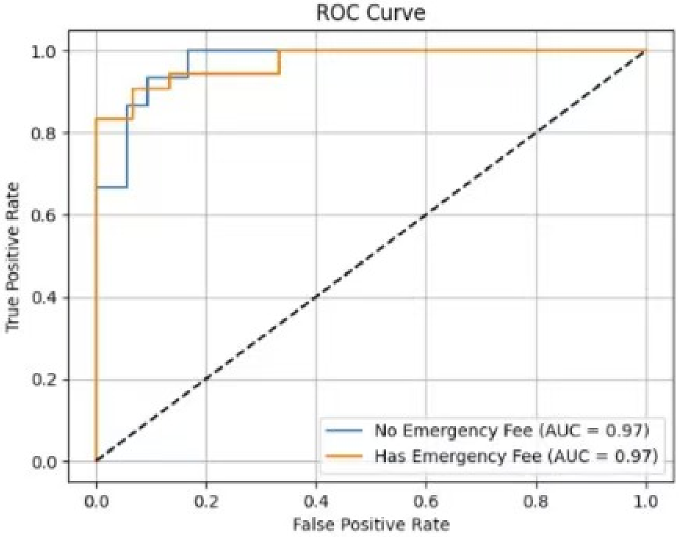


Figure 5.5: ROC Curve for *HasEmergencyFee* Predictions.

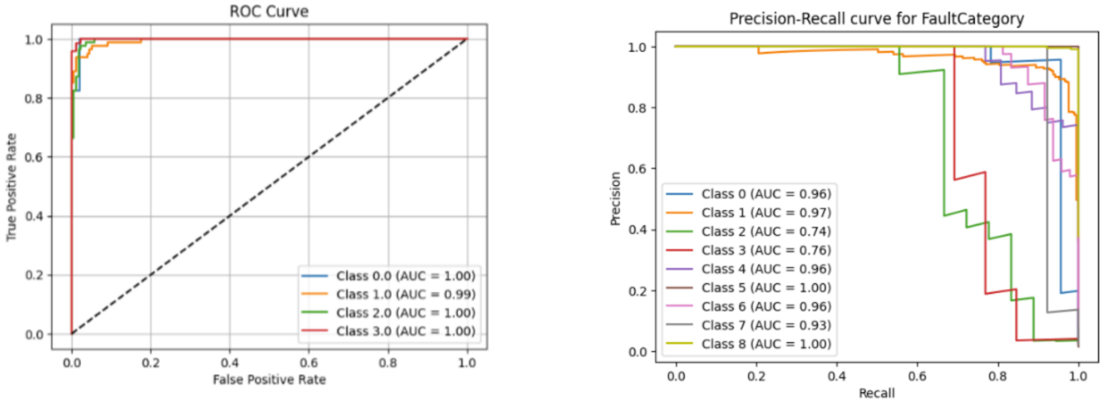
These results confirm that the NRP-GCN model is highly effective across binary and multiclass classification tasks, consistently achieving strong separation between classes.

Validation Curves

Validation curves were generated by varying the number of decision trees in the ensemble models. The results indicate:

- Training accuracy remains consistently high across all tree counts, demonstrating the model's capability to fully capture patterns in the training data.
- Cross-validation accuracy increases significantly when the tree count is below 50, but shows diminishing returns beyond 300 trees, stabilizing near 0.9 for both targets.

These observations suggest that increasing the number of trees beyond a certain threshold does not substantially enhance model performance, and careful tuning is essential to balance computational cost and predictive accuracy.



(a) ROC Curve for *EmergencyFeeCategory* Predictions.

(b) ROC Curve for *FaultCategory* Predictions.

Figure 5.6: ROC Curves for Predictions of *EmergencyFeeCategory* and *FaultCategory*.

Performance and Recommendations

The shaded regions in the learning curves illustrate the confidence intervals of the cross-validation results. As the training dataset grows, the variance decreases, reflecting improved stability in the model's predictions. However:

- For further enhancement, additional samples or data augmentation techniques could be explored to reduce the generalization gap.
- Based on the validation curves, the optimal number of trees in the ensemble models should be capped at 300 to maximize efficiency without compromising accuracy.

5.5 Risk Statement Generation

In addition to predicting `EmergencyFeeCategory`, `HasEmergencyFee`, and `FaultCategory`, we also generate risk statements that assess the likelihood of increased emergency fees based on fault-related features. These statements help identify faults that are more likely to result in higher emergency costs.

For each fault code, a risk score is calculated based on several features, including the fault code itself, its associated proportions and counts, as well as geographical data such as latitude and longitude. If the predicted risk score exceeds a certain threshold (T), the model generates a risk statement.

The generated risk statement is structured as follows:

- The ratio of combination `639-2` is 0.42857142857142855, and the risk of first aid costs increases when the number of times is 3.
- The ratio of combination `3516-2` is 0.2857142857142857. When the number of times is 2, the risk of first aid costs increases.
- The ratio of combination `639-5` is 0.14285714285714285. When the number of times is 1, the risk of first aid costs increases.
- The ratio of combination `904-8` is 8.14285714285714285, and the risk of first aid costs increases when the number of times is 1.
- The ratio of combination `518114-5` is 0.3793103448275862, and the risk of first aid costs increases when the number is 11.

By generating these risk statements, the model provides actionable insights into which faults are more likely to lead to increased emergency fees, helping decision-makers to better manage and mitigate potential risks.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

This work presents a hybrid modeling approach that integrates Graph Neural Networks (GNNs) with traditional machine learning models to enhance prediction accuracy for tasks involving both graph-structured and tabular data. The hybrid model leverages the strengths of GNNs in capturing relational and structural information while utilizing the robustness and efficiency of traditional models for learning non-linear relationships in tabular features.

The key contributions and findings of this work are summarized as follows:

- **Graph Data Construction:** The data was transformed into a graph structure where nodes represent vehicle-specific attributes (e.g., geographical location, emergency fee status, and fault categories) and edges capture relational dependencies based on fault combinations and weighted relationships. This graph representation enriched the dataset with contextual information not readily available in tabular format.

- **Graph Neural Network Embeddings:** GNN embeddings were extracted using a multi-layer Graph Convolutional Network (GCN). These embeddings encapsulated both the node-specific features and their neighborhood relationships, providing a robust representation for downstream tasks.

- **Fusion of Graph and Tabular Features:** The GNN embeddings were combined with tabular features, such as fault proportions and counts, to create a unified feature set. This integration effectively bridged the gap between graph-based and tabular data, capturing complementary information from both domains.

- **Traditional Model Ensemble:** The unified feature set was input into an ensemble of traditional machine learning models, including Random Forests, XGBoost, and Decision Trees. The ensemble approach leveraged the strengths of individual models, improving classification performance across targets such as *HasEmergencyFee*, *EmergencyFeeCategory*, and *FaultCategory*.

- **Sample Imbalance Handling:** The application of SMOTEENN ensured balanced training data, reducing biases in predictions and improving performance for minority classes.

The hybrid model achieved impressive results across all evaluated targets:

- **HasEmergencyFee:** Demonstrated a cross-validation accuracy of 97.01% with a standard deviation of 3.65%, indicating stable predictions for emergency fee presence.

- **EmergencyFeeCategory:** Achieved a cross-validation accuracy of 94.96% with a low standard deviation of 2.02%, reflecting consistent classification performance across fee categories.

- **FaultCategory:** Attained near-perfect cross-validation accuracy of 98.04% with a minimal standard deviation of 0.74%, showcasing the model's robustness in fault type prediction.

6.2 Future Work

While the proposed hybrid model demonstrates significant potential, there remain several avenues for future research to further improve and extend this approach:

- **Dynamic Graph Construction:** The current graph is constructed statically based on predefined relationships. Future work can explore dynamic graph construction methods that adaptively learn relationships based on data features and time-dependent interactions.

- **Explainability and Interpretability:** Although the model performs well, the interpretability of GNN embeddings and their influence on predictions remains an open challenge. Future efforts could focus on developing explainable GNN techniques to provide insights into model decisions.

- **Incorporation of Temporal Features:** Vehicle fault data often exhibits temporal dependencies. Incorporating time-series modeling techniques, such as Temporal GNNs or Long Short-Term Memory networks (LSTMs), could improve predictions by leveraging temporal patterns.

- **Scalability to Larger Graphs:** As the dataset size grows, the computational cost of graph construction and GNN training increases. Future work could investigate scalable GNN architectures and graph sampling methods to efficiently handle larger datasets.

- **Integration with Real-Time Systems:** Extending this hybrid approach to real-time vehicle health monitoring and predictive maintenance systems would require optimizing inference speed and integrating with vehicle diagnostic tools.

References

[1] Minghu Zhang, Jianwen Guo, Xin Li, and Rui Jin, ”Data-Driven Anomaly Detection Approach for Time-Series Streaming Data,” Key Laboratory of Remote Sensing of Gansu Province, Northwest Institute of Eco-Environment and Resources, Chinese Academy of Sciences, Lanzhou, China, 2020.

[2] Chuxu Zhang, Dongjin Song, Yuncong Chen, Xinyang Feng, Cristian Lumezanu, Wei Cheng, Jingchao Ni, Bo Zong, Haifeng Chen, and Nitesh V. Chawla, ”A Deep Neural Network for Unsupervised Anomaly Detection and Diagnosis in Multivariate Time Series Data,” University of Notre Dame, IN, USA; NEC Laboratories America, NJ, USA; Columbia University, NY, USA, 2020.

[3] S. Neupane, I. A. Fernandez, W. Patterson, S. Mittal, and S. Rahimi, ”A Temporal Anomaly Detection System for Vehicles Utilizing Functional Working Groups and Sensor Channels,” in *2022 IEEE 8th International Conference on Collaboration and Internet Computing (CIC)*, IEEE, pp. 24, 2022.

[4] M. C. Moura, E. Zio, I. D. Lins, and E. Droguett, ”Failure and Reliability Prediction by Support Vector Machines Regression of Time Series Data,” Department of Production Engineering, Federal University of Pernambuco, Brazil; Department of Energy, Polytechnic of Milan, Italy; Ecole Centrale Paris et Supelec, France.

[5] L. Biddle and S. Fallah, ”A Novel Fault Detection, Identification and Prediction Approach for Autonomous Vehicle Controllers Using SVM,” Connected Autonomous

Vehicles Lab (CAV-Lab), Department of Mechanical Engineering Sciences, University of Surrey, 2021.

[6] A. Theissler, ”Detecting Known and Unknown Faults in Automotive Systems Using Ensemble-Based Anomaly Detection,” Faculty of Information Technology, Esslingen University of Applied Sciences, 2017.

[7] C. Zhang, D. Song, Y. Chen, X. Feng, C. Lumezanu, W. Cheng, J. Ni, B. Zong, H. Chen, and N. V. Chawla, ”A Deep Neural Network for Unsupervised Anomaly Detection and Diagnosis in Multivariate Time Series Data,” in *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence (AAAI-19)*, 2019.

[8] K. Choi, J. Yi, C. Park, and S. Yoon, ”Deep Learning for Anomaly Detection in Time-Series Data: Review, Analysis, and Guidelines,” *IEEE Access*, vol. 9, pp. 120020–120035, 2021.

[9] L. Biddle and S. Fallah, ”A Novel Fault Detection, Identification and Prediction Approach for Autonomous Vehicle Controllers Using SVM,” Connected Autonomous Vehicles Lab (CAV-Lab), Department of Mechanical Engineering Sciences, University of Surrey, 2021.

[10] S. M. Namburu, M. Wilcutts, S. Chigusa, L. Qiao, K. Choi, and K. Pattipati, ”Systematic Data-Driven Approach to Real-Time Fault Detection and Diagnosis in Automotive Engines,” Toyota Motor Engineering Manufacturing North America, USA; Department of ECE, University of Connecticut, USA.

[11] J. Wang, C. Zhang, X. Ma, Z. Wang, Y. Xu, and R. Cattley, ”A Multivariate Statistics-Based Approach for Detecting Diesel Engine Faults with Weak Signatures,” College of Power and Energy Engineering, Harbin Engineering University, China; Centre for Efficiency and Performance Engineering, University of Huddersfield, UK, 2020.

[12] L.-J. Cao and F. E. H. Tay, ”Support Vector Machine with Adaptive Parameters in Financial Time Series Forecasting,” *IEEE Transactions on Neural Networks and Learning Systems (TNNLS)*, vol. 14, no. 6, pp. 1506–1518, 2003.

[13] Y. Xia and J. Chen, ”Traffic Flow Forecasting Method Based on Gradient Boosting Decision Tree,” in *FMSMT*, Atlantis Press, pp. 413–416, 2017.

[14] B. Biller and B. L. Nelson, ”Modeling and Generating Multivariate Time-Series Input Processes Using a Vector Autoregressive Technique,” *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, vol. 13, no. 3, pp. 211–237, 2003.

[15] G. E. Box and D. A. Pierce, ”Distribution of Residual Autocorrelations in Autoregressive-Integrated Moving Average Time Series Models,” *Journal of the American Statistical Association (JASA)*, vol. 65, no. 332, pp. 1509–1526, 1970.

[16] M. Jin, Y. Zheng, Y.-F. Li, S. Chen, B. Yang, and S. Pan, ”Multivariate Time Series Forecasting with Dynamic Graph Neural ODEs,” *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, 2022.

[17] B. Zhao, H. Lu, S. Chen, J. Liu, and D. Wu, ”Convolutional Neural Networks for Time Series Classification,” *Journal of Systems Engineering and Electronics*, vol. 28, no. 1, pp. 162–169, 2017.

[18] J. T. Connor, R. D. Martin, and L. E. Atlas, ”Recurrent Neural Networks and Robust Time Series Prediction,” *IEEE Transactions on Neural Networks and Learning Systems (TNNLS)*, vol. 5, no. 2, pp. 240–254, 1994.

[19] Q. Wen, T. Zhou, C. Zhang, W. Chen, Z. Ma, J. Yan, and L. Sun, ”Transformers in Time Series: A Survey,” in *Proceedings of IJCAI*, 2023.

[20] G. Jin, Y. Liang, Y. Fang, J. Huang, J. Zhang, and Y. Zheng, ”Spatio-Temporal Graph Neural Networks for Predictive Learning in Urban Computing: A Survey,” *arXiv preprint*, vol. abs/2303.14483, 2023.

[21] Min Wang, Xijun Zhu, “Association Rule Analysis Based on Improved Apriori Algorithm,” *School of Information Science and Technology, Qingdao University of Science and Technology*, Qingdao, Shandong, China, Received: May 18, 2021; Accepted: June 15, 2021; Published: June 22, 2021.

Appendix I: Contribution of Responsibilities

Team Member	**Tasks**
Wu You	• Conducted literature review • Performed data cleaning and transformation • Computed association analysis • Conducted intermediate model testing • Constructed graph data and developed hybrid NRP-GCN model • Conducted model validation • Authored the paper • Prepared presentation materials (PPT)

Zhang Sirui	• Conducted literature review • Processed and transformed geographical coordinate data • Conducted and improved association analysis and provided theoretical explanations • Implemented one-hot encoding and developed Random Forest and SVM models • Conducted model validation • Authored the paper • Prepared presentation materials (PPT)
Su Junwei	• Conducted literature review • Processed NLP-based category data • Constructed and tested intermediate models • Conducted model validation • Authored the paper • Prepared presentation materials (PPT)

Appendix II: Nine Fault Classification Results

Rear Axle (7 Faults)

- Abnormal noise from the main reducer assembly of the rear axle
- Cracked left housing of the planetary gear differential in the rear axle
- Sand hole in the rear axle reducer housing
- Poor welding of the rear axle shell rear cover
- Burnt cross shaft assembly of the rear axle driveshaft
- Abnormal wear of the planetary gear support washer in the rear axle
- Fracture of the rear axle driveshaft tube

Engine (19 Faults)

- Failure of the oil pressure sensor assembly
- Loosening of the oil pump outlet pipe assembly
- Damage to the oil pump outlet pipe assembly
- Oil leakage from the engine oil cooler assembly
- Functional failure of the oil pump assembly
- Abnormal function of the oil pump assembly
- Damage to the oil inlet pipe assembly
- Damage to the oil collector assembly

- Oil leakage from the oil pump assembly
- Poor sealing of the oil pump assembly
- Deformation of the oil suction pipe
- Damage to the oil pressure sensor assembly
- Abnormal electrical signal from the oil pressure sensor assembly
- Oil leakage from the engine assembly
- Short circuit in the engine wiring harness assembly
- Software failure in the engine control unit assembly
- Poor welding of the oil suction pipe
- Abnormal wear of the oil pump assembly
- Software failure in the engine drive clutch assembly

Transmission (23 Faults)

- Oil leakage from the clutch master-to-slave pump front oil pipe assembly
- Abnormal noise in the rear sub-box assembly (gearbox)
- Air leakage from the clutch booster assembly
- Abnormal function of the clutch cover and pressure plate assembly
- Failure of the clutch booster assembly
- Sticking of the gearbox air control valve assembly
- Sticking of the pull-type clutch release bearing assembly

- Cracking of the pull-type clutch release bearing assembly
- Functional failure of the gearbox neutral switch assembly
- Oil leakage from the clutch booster assembly
- Fracture of the driveshaft spline shaft fork assembly
- Abnormal noise in the steering drive device with an adjuster assembly
- Abnormal wear of the driveshaft spline shaft fork assembly
- Functional failure of the steering drive device with an adjuster assembly
- Loosening of the pull-type clutch release bearing assembly
- Sticking of the clutch booster assembly
- Damage to the clutch release bearing seat and bearing assembly
- Sticking of the gearbox internal control system assembly
- Air leakage from the AMT automatic transmission with retarder assembly
- Fracture of the double-head stud (clutch)
- Functional failure of the fan clutch with a viscous fan assembly
- Oil leakage from the clutch master cylinder with reservoir assembly
- Vibration in the intermediate driveshaft and support assembly

Fuel Supply System (19 Faults)

- Functional failure of the urea pump motor assembly
- Oil leakage from the suction pipe assembly
- Blockage of the injector assembly
- Oil leakage from the fuel injection pump assembly
- Cracking of the urea supply hose assembly (quantitative valve to injector)
- Oil leakage from the inlet pipe assembly (filter to fuel injection pump)
- Abnormal function of the fuel transfer pump assembly
- Failure of the electric fuel pump assembly
- Blockage of the inlet hose assembly (pump to steering gear)
- Damage to the timing belt pulley (high-pressure fuel pump)
- Optimization of urea pump crystallization in some E298/E297 vehicles
- Oil leakage from the fine fuel filter assembly
- Functional failure of the oil pump system assembly
- Oil leakage from the fuel tank assembly
- Blockage of the after-treatment hydrocarbon injector
- Oil leakage from the inlet hose assembly (pump to steering gear)
- Oil leakage from the steering oil pump and gear assembly
- Oil leakage from the steering oil pump assembly
- Circuit break in the urea supply hose assembly (metering pump to injector)

Cooling and Radiator System (6 Faults)

- Blockage in the EGR cooler assembly
- Water leakage from the water pump assembly
- Water leakage from the thermostat assembly
- Water leakage from the radiator assembly
- Damage to the EGR cooler assembly
- Functional failure of the thermostat assembly

Brake System (6 Faults)

- Functional failure of the brake light switch assembly
- Cracking of the rear right spring brake chamber
- Air leakage from the brake valve assembly
- Damage to the rear air chamber bracket
- Cracking of the rear brake drum
- Air leakage from the rear right spring brake chamber

After-Treatment System (8 Faults)

- Damage to the differential pressure sensor assembly (after-treatment ends)
- Blockage in the DPF diesel particulate filter sub-assembly
- Cracking of the after-treatment assembly

- Blockage of the after-treatment assembly
- Blockage of the after-treatment and installation process assembly
- Failure of the differential pressure sensor assembly (after-treatment ends)
- Abnormal function of the after-treatment assembly
- Damage to the after-treatment assembly

Electronic Control System (4 Faults)

- Software failure in the EECU electronic control unit assembly
- Abnormal noise from the generator belt tensioner assembly
- Software failure in the after-treatment control unit (DCU)
- Functional failure of the electronic odometer sensor assembly

Others (142 Faults)

Full list omitted for brevity.

- Communication terminal assembly failure in vehicle networking
- Abnormal wear in lower connecting rod bearings
- O-ring damage
- Air leakage in the differential valve assembly
- Vibration in the driver’s side external mirror and bracket assembly
- ...