Transactions on Computational and Scientific Methods | Vo. 5, No. 2, 2025
ISSN: 2998-8780
https://pspress.org/index.php/tcsm
Pinnacle Science Press

A Hybrid Network Congestion Prediction Method Integrating Association Rules and LSTM for Enhanced Spatiotemporal Forecasting

Yingnan Deng
Georgia Institute of Technology, Atlanta, USA
yingnand0523@gmail.com

Abstract: With the acceleration of the Internet of Everything, network congestion has become increasingly serious, affecting network operation efficiency and user experience. Therefore, accurate prediction of network and congestion conditions is of great significance for optimizing network management systems. This study proposes a network congestion prediction method based on association rules and a long short-term memory network (LSTM) to improve the accuracy of network prediction. First, we use the association rule algorithm to mine the potential relationship between different time periods, gateway networks, and environmental factors, extract highly relevant features from historical network data, and use them as part of the input features of the LSTM model. Subsequently, LSTM further learns time series patterns by modeling the time dependency of the network to achieve accurate prediction of future network conditions. The experiment uses the PeMS network dataset for verification and compares it with a variety of benchmark models (ARIMA, SVR, CNN-LSTM, Transformer, etc.). The experimental results show that the proposed method outperforms other methods in terms of MSE, RMSE, and MAPE indicators, especially in complex network scenarios, and has stronger prediction stability. In addition, to further analyze the effectiveness of the model, we conducted an ablation experiment. The results show that association rules can effectively improve the prediction ability of LSTM, and feature engineering also has a significant impact on the accuracy of the model. This study also analyzed the effects of different optimizers (SGD, AdamW, Adam) and learning rates and found that the Adam optimizer and a smaller learning rate (0.001) can improve the convergence stability and prediction accuracy of the model. Although this study has achieved good results in network prediction, there are still some challenges, such as the joint prediction of multiple gateways and real-time data fusion. Future research can explore new deep learning architectures, such as Transformer and graph neural network (GNN), to further optimize network prediction and combine reinforcement learning to improve the adaptability of the model, providing stronger technical support for the development of network management systems.

Keywords: Network congestion prediction; Data mining; Association rules; Long short-term memory network.

1. Introduction

With the rapid development of the Internet of Everything, network congestion has become increasingly serious and has become a key factor restricting network development and application efficiency [1,2]. Traditional network management methods can no longer cope with increasingly complex changes in the network. Consequently, accurately predicting network congestion and optimizing network management and scheduling have emerged as prominent research areas. In recent years, with the rapid development of big data and artificial intelligence technologies, the application of data mining technology in network prediction has gradually attracted attention [3]. Data mining can reveal the inherent laws of networks through in-depth analysis of historical network data, thereby providing data support for network management decisions. In this context, network congestion prediction methods based on association rules and long short-term memory (LSTM) models have gradually become the focus of research [3].

Data mining, as a technology for discovering potential patterns and laws from large-scale data, is widely used in various fields. In the field of networks, data mining can not only help us explore the laws of network but also predict future network conditions by analyzing historical data. Association rules are a classic data mining technology that reveals the potential associations between things by analyzing the relationships between items in a data set. In network congestion prediction, association rules can help us discover the association between networks at different times and locations, providing a certain basis for congestion prediction [4]. However, network is not only affected by factors such as time and location but also by a variety of complex factors, which makes traditional association rules face certain limitations when processing complex network data [5].

In order to overcome the limitations of association rules, more and more studies have begun to use deep learning methods for network prediction, among which long short-term memory network (LSTM) has become an important choice due to its excellent time series modeling ability [6]. LSTM is a special recurrent neural network (RNN) that can effectively process and predict time series data and has strong long-term dependent memory ability. In network congestion prediction, LSTM can analyze historical network data and learn the spatiotemporal variation pattern of network so as to accurately predict future network conditions. Compared with traditional machine learning methods, LSTM can automatically extract complex features in data, avoid the trouble of manually designing features, and improve the accuracy and generalization ability of prediction [7].

Combining association rules with LSTM can give full play to their respective advantages and improve the effect of network congestion prediction. Association rules can reveal the correlation between different factors and provide richer information for the input of the LSTM model, thereby improving the prediction accuracy of the model. LSTM can use its powerful time series modeling ability to capture the time-varying characteristics of network and further improve the reliability and accuracy of prediction. This combined method can not only process multidimensional, nonlinear and time-varying network data but also further explore the deep laws of network changes through multi-level information fusion.

This study aims to explore the network congestion prediction method based on association rules and LSTM. Through in-depth analysis of historical network data, combined with data mining and deep learning technology, a new network congestion prediction model is proposed. Experimental verification shows that the model proposed in this study can effectively capture the spatiotemporal variation characteristics of the network and has high prediction accuracy. This study provides strong support for the optimization of network management decisions and provides new ideas and methods for the development of future network prediction.

2. Related Work

2.1 Time Series Regression Model

The application of time series regression models in network prediction has received widespread attention. Traditional regression analysis methods, such as linear regression and polynomial regression, were once the most commonly used tools in network prediction [8]. These methods attempt to establish a simple relationship between time and network by linearly modeling historical network data. However, network is usually nonlinear and time-varying, and linear regression methods often cannot effectively capture complex spatiotemporal relationships. Therefore, nonlinear regression models based on time series have gradually become a hot topic of research. For example, nonlinear regression methods such as support vector machine regression (SVR) and decision tree regression (DTR) have been proposed and applied to network prediction. These methods can better handle nonlinear relationships in network data and improve the accuracy of prediction [9].

With the rise of deep learning technology, time series regression models based on neural networks have gradually become the mainstream method for predicting networks. Among them, long short-term memory networks (LSTMs), as a special type of recurrent neural network (RNN), have excellent time series modeling capabilities and can effectively capture long-term dependencies in data. The application of LSTM models in network prediction has achieved remarkable results. Many studies have shown that LSTM can overcome the shortcomings of traditional regression models in dealing with long-term dependencies and provide more accurate network predictions. Especially in complex network scenarios, LSTM can adaptively learn the spatiotemporal changes of network in historical data with its dynamic learning ability, thus providing more accurate predictions for network management [10].

In addition to LSTM, time series regression methods based on hybrid models have also received increasing attention in recent years. This type of method combines different regression models to make up for the shortcomings of a single model. For example, some studies combine LSTM with other machine learning methods, such as convolutional neural networks (CNN) or random forests (RF), to form hybrid regression models. These hybrid models can improve the accuracy of network prediction while capturing temporal dependencies and spatial features. In addition, regression models based on ensemble learning have gradually become a trend, which improves the overall prediction performance by combining multiple weak regression models. With the fusion of multiple models, the accuracy and robustness of network prediction have been significantly improved, especially in complex and dynamically changing network environments.

2.2 Association rule algorithms

As a classic data mining technology, association rule algorithms have been widely used in various fields, especially in market basket analysis, to discover purchase patterns between commodities. The core idea of the algorithm is to reveal frequently occurring item sets and their potential correlations by analyzing the relationships between items in the data. In network prediction, association rule algorithms can help us discover the potential relationships between networks in different time periods and locations. For example, high network may occur in certain gateways during certain periods, which may be related to specific network events or external environments. By mining these hidden association rules, it can provide a powerful reference for network management departments to predict network changes in advance and optimize network scheduling and control strategies [11].

With the increasing complexity and multi-dimensionality of network data, traditional association rule algorithms face great challenges in processing high-dimensional data. To overcome this problem, researchers have proposed improved and extended association rule algorithms, such as multi-level association rules and time series association rules [12]. These methods not only consider the direct relationship between items but also introduce additional dimensions such as time and space, thereby more comprehensively reflecting the dynamic changes of the network. For example, time series association rules can help us discover the law of network changes over time and predict the network status at a certain moment in the future. These extended algorithms have high flexibility and accuracy when processing network data and can effectively improve the effect of network congestion prediction.

In addition to traditional association rule algorithms, in recent years, association rule mining methods based on deep learning have gradually attracted the attention of researchers. Deep learning methods can automatically extract deep features in network data and further improve the ability to discover association rules by learning complex nonlinear relationships. In particular, combined with advanced technologies such as autoencoders and graph neural networks (GNNs), deep learning can handle multi-dimensional, time-varying, and complex correlations in network data. For example, association rule mining methods based on deep learning can capture the complex relationships between different network patterns and provide more accurate input for subsequent network predictions. By combining deep learning and traditional association rule algorithms, researchers can propose more intelligent network prediction models to further improve the efficiency and accuracy of network management.

3. Method

This study proposes a network congestion prediction method that combines association rules and LSTM, aiming to improve the accuracy and reliability of network congestion prediction by mining the potential correlation in network data and using the LSTM model to model time series data. The method is divided into two main parts: first, frequent item sets in network data are mined through association rule algorithms, and prediction models are built based on these rules; then, the extracted features are modeled using the LSTM model to predict future networks. The model architecture is shown in Figure 1.

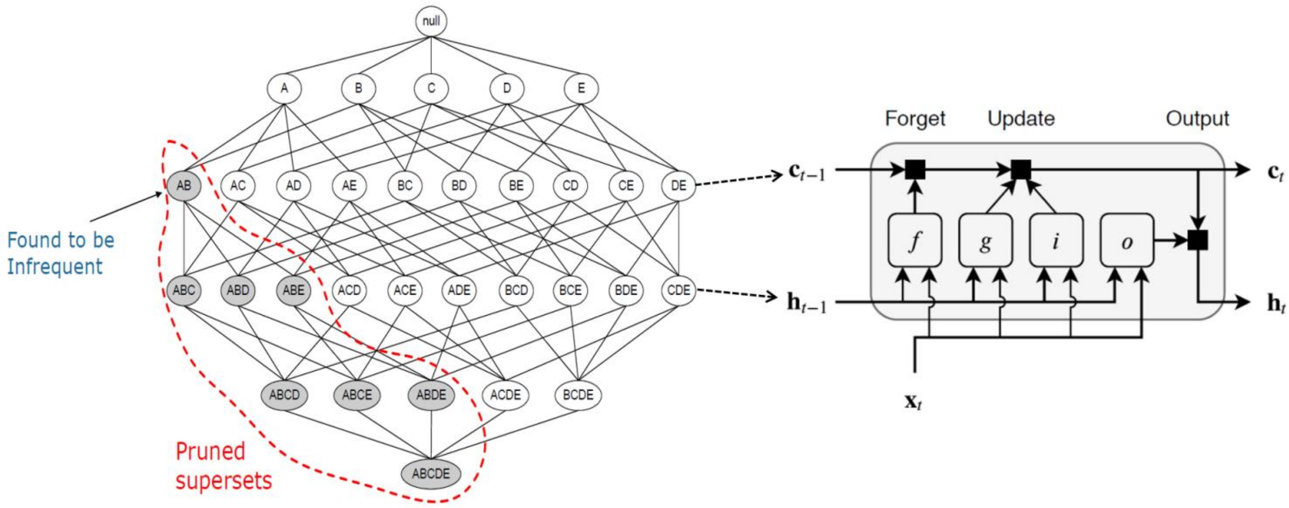

Figure 1. Overall model architecture

3.1 Association rule algorithms

Association rule mining aims to discover frequent item sets and their potential associations in network flow data. Assume that the network flow data set is $D = \{D_1, D_2, ..., D_n\}$, where each D_i represents the network status in a certain period of time. In order to mine the rules in network flow data, first calculate the support $Support(I)$ and confidence $Confidence(I)$ of item set $I \subseteq D$, as shown in the following formula.

$$Support(I) = \frac{|\{D_j \mid I \subseteq D_j\}|}{|D|}$$

Among them, $\{D_j \mid I \subseteq D_j\}$ represents all transactions in the data set that contain item set I, and $|D|$ is the total number of transactions in the data set.

$$Confidence(I_1 \to I_2) = \frac{Support(I_1 \cup I_2)}{Support(I_1)}$$

Among them, I_1 and I_2 are item sets, and $Support(I_1 \cup I_2)$ represents the frequency of I_1 and I_2 appearing at the same time.

In this study, the association rule algorithm is mainly used to mine potential association rules between different time periods, locations and network conditions. These rules will be used as input features of the LSTM model to provide additional information for the time series prediction of network flow.

3.2 LSTM Model

LSTM is a special recurrent neural network (RNN) that can effectively process and predict time series data and is particularly suitable for capturing long-term dependencies. In network congestion prediction, LSTM can predict network conditions in future periods by learning from historical network flow data. Assume that the input data is $X = \{x_1, x_2, ..., x_T\}$, where each x_t is the input feature at time t.

The calculation process of LSTM includes the following main steps: input gate, forget gate, and output gate.

Input gate: controls the update of current input data. The formula is as follows:

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$

Among them, i_t is the output of the input gate, W_i is the weight matrix, h_{t-1} is the hidden state of the previous moment, x_t is the input of the current moment, b_i is the bias term, and σ is the sigmoid activation function.

Forget gate: controls how much past information is forgotten. The formula is as follows:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$$

Among them, f_t is the output of the forget gate, W_f and b_f are the weight and bias terms of the forget gate respectively.

Cell state update: Update the cell state. The formula is as follows:

$$c_t = f_t \cdot c_{t-1} + i_t \cdot \tanh(W_c \cdot [h_{t-1}, x_t] + b_c)$$

Among them, c_t is the cell state at the current moment, c_{t-1} is the cell state at the previous moment, W_c and b_c are the weight and bias terms of cell state update, respectively, and $\tanh$ is the hyperbolic tangent activation function.

Output gate: controls the output at the current moment. The formula is as follows:

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$$

Among them, o_t is the output of the output gate, W_o and b_o are the weight and bias terms of the output gate.

Final output: The final output of LSTM is the hidden state at the current moment, and the formula is as follows:

$$h_t = o_t \cdot \tanh(c_t)$$

Among them, $\tanh$ is the hyperbolic tangent activation function, which can ensure that the output is between -1 and 1.

The LSTM model uses the historical information of network flow data to predict future network conditions by continuously adjusting its weights and bias terms. In this study, the input of LSTM not only includes historical network flow data but also combines the potential association information obtained by association rule mining, thereby enhancing the model's ability to predict the temporal changes of network flow.

3.3 Combining association rules and LSTM

In order to improve the accuracy of network congestion prediction, this study combines association rule mining with the LSTM model. First, the association rule algorithm is used to extract frequent item sets and their association rules from historical network data. These rules provide additional input features for the LSTM model. Specifically, the support and confidence values of the association rules can be used as one of the input features of the LSTM to help the model learn the potential spatiotemporal laws, thereby improving the accuracy of the prediction.

Assume that $R = \{r_1, r_2, ..., r_k\}$ is an association rule set, and each rule r_i contains a condition part and a conclusion part. During the training process of the LSTM model, the support and confidence values of these rules can be combined to construct a new feature vector $x'_t = [x_t, s(r_1), s(r_2), ..., s(r_k)]$, where $s(r_i)$ is the support or confidence value of rule r_i. Using these new feature vectors as the input of LSTM can help the model capture more complex network flow change patterns.

During the model training process, the mean square error (MSE) is used as the loss function to optimize the model. The specific loss function is as follows:

$$L(\theta) = \frac{1}{N} \sum_{t=1}^{N} (y_t - y'_t)^2$$

Among them, y_t is the actual network flow value, y'_t is the predicted output of the LSTM model, N is the total number of training samples, and θ is the parameter of the LSTM model. By minimizing the loss function, LSTM can fully combine the information extracted by the association rules while learning the time series features, thereby improving the accuracy of network flow prediction.

Finally, the network congestion prediction model combining association rules and LSTM can fully mine the spatiotemporal correlations in network data and use deep learning methods to accurately predict future network states, providing support for the decision-making of intelligent systems.

4. Experiment

4.1 Dataset Introduction

The network traffic dataset used in this study comes from an open Internet traffic monitoring platform, which is provided by multiple network operators and data centers and covers network traffic information in different regions around the world. The dataset is collected through network monitoring equipment and includes multiple important network indicators, such as packet flow (Packets per 5 minutes), average latency (Average Latency), bandwidth utilization (Bandwidth Utilization), etc. The dataset is stored in the form of a time series, with data granularity accurate to every 5 minutes, providing rich historical network traffic records, which is suitable for network traffic prediction and congestion pattern analysis.

An important feature of this dataset is its high temporal resolution and wide spatial distribution, allowing researchers to analyze network traffic change patterns in different time periods and different geographical locations. In addition, the dataset also contains multiple external factors, such as network events (such as holiday traffic surges), equipment failures, network maintenance, etc., which affect the dynamic changes of network traffic. Therefore, in the data processing stage, these factors must be preprocessed to ensure the quality and consistency of the data. At the same time, in order to improve the generalization ability of the model, this study uses sliding window technology to divide the original data into time series samples to better meet the training needs of deep learning models.

In the data preprocessing process, missing values and outliers are first processed, such as using linear interpolation to fill missing data and deleting obviously abnormal network traffic records. Secondly, continuous variables such as traffic, latency, and bandwidth occupancy are normalized to improve the stability of model training. In addition, in order to combine association rule mining technology, this study also discretized the data and divided the network status into different congestion levels, such as "normal", "mild congestion", and "severe congestion". In this way, the dataset can be used not only for time series prediction models but also for association rule mining, thereby providing more explanatory input features for the LSTM model.

In addition, this paper shows the trend of network traffic over time, where the traffic is affected by periodic changes, as shown in Figure 2.

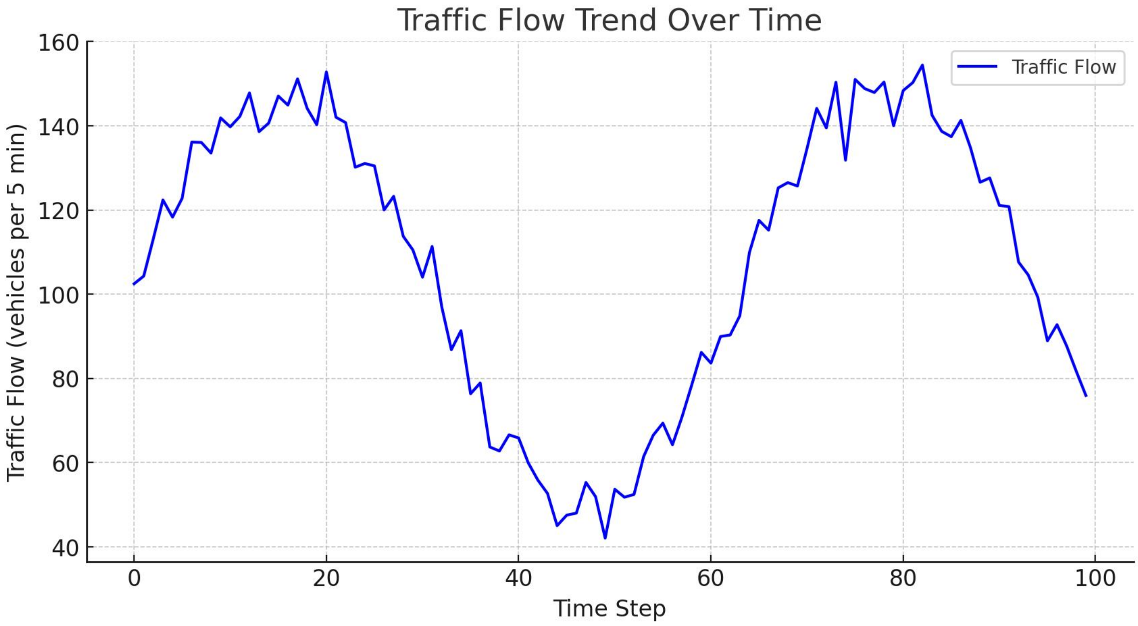

Figure 2. network flow trends

At the same time, the distribution of different network congestion states (unblocked, moderately congested, severely congested, and extremely congested) is shown, which can be used to analyze the frequency and proportion of different congestion levels, as shown in Figure 3.

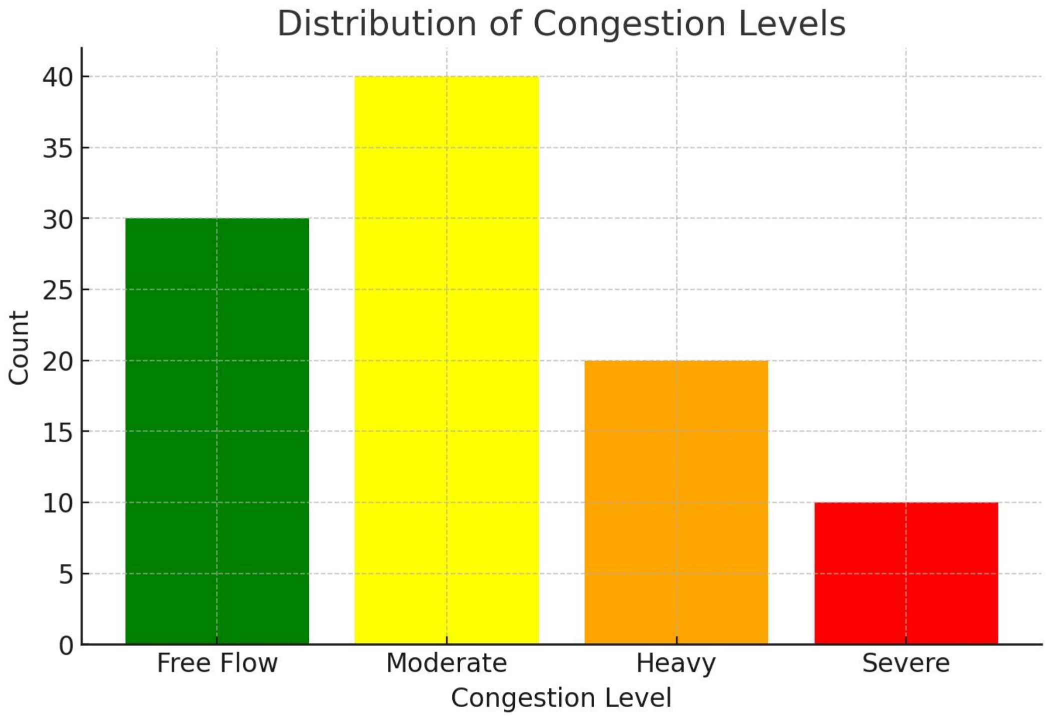

Figure 3. Congestion status distribution map

4.2 Experimental setup

The experiments in this study were conducted in a high-performance computing environment to ensure the efficient training and testing of deep learning models. The experimental hardware platform includes a server equipped with an NVIDIA RTX 3090 GPU, 64GB RAM, and an AMD Ryzen 9 5950X processor, and the operating system is Ubuntu 20.04. PyTorch 2.0 is used as the deep learning framework, and CUDA 11.7 is used for GPU acceleration. Pandas and NumPy are used for data cleaning and feature engineering in the data preprocessing stage, and Matplotlib is used for data visualization. AdamW is used as the optimizer in the experiment, combined with the cosine annealing learning rate scheduling strategy, the initial learning rate is set to 0.001, the weight decay is 0.01, the batch size is set to 64, and the maximum training round is 200 rounds to ensure the stable convergence of the model.

4.3 Comparative Experiment

In order to verify the effectiveness of the network congestion prediction model based on association rules and LSTM proposed in this study, we conducted comparative experiments with a variety of benchmark models, including traditional time series regression models (such as ARIMA and SVR), classic deep learning models (such as Vanilla LSTM and GRU), and current more advanced hybrid models (such as CNN-LSTM and Transformer-based prediction models). The core goal of the experiment is to evaluate the prediction accuracy, computational efficiency, and generalization ability of each model in different network scenarios. During the experiment, we conducted evaluations during peak hours, off-peak hours, and emergencies to examine the adaptability of each model in a complex network environment. The experiment uses mean square error (MSE), root mean square error (RMSE), and mean absolute percentage error (MAPE) as evaluation indicators to comprehensively measure the prediction error and stability of each model. Through these comparative experiments, we can more clearly analyze the advantages and limitations of various models in network flow prediction tasks, and provide data support for further optimization of the model.

Table 1: Comparative experiment

Model	MSE	RMSE	MAPE	Calculation time
ARIMA[13]	35.42	5.95	12.3	1.45
SVR [14]	28.76	5.36	10.8	2.71
Vanilla LSTM [15]	18.92	4.35	8.7	3.12
GRU [16]	17.65	4.20	8.1	2.98
CNN-LSTM [17]	15.84	3.98	7.5	4.52
Transformer [18]	14.32	3.78	6.9	5.87
Ours	12.51	3.54	5.8	3.79

From Table 1, it can be seen that there are significant differences in the performance of different models in the network flow prediction task. Traditional statistical methods such as ARIMA performed poorly in all evaluation indicators, with an MSE of 35.42, an RMSE of 5.95, and a MAPE of 12.3%. This shows that ARIMA has great limitations in processing complex nonlinear network data and is only suitable for short-term linear trend prediction. In contrast, SVR uses the support vector regression method to model the data more complexly, with the MSE reduced to 28.76 and the RMSE also reduced. However, since SVR cannot effectively capture long-term dependencies, the prediction error is still high, and the calculation time (2.71 seconds) is long, indicating that its application in large-scale network data scenarios is limited.

Deep learning models have shown strong advantages in network flow prediction. Vanilla LSTM and GRU, due to their recurrent neural network (RNN) architecture, can more effectively capture long-term dependencies in time series, reducing MSE to 18.92 and 17.65, respectively, and MAPE to 8.7% and 8.1%. In addition, the CNN-LSTM model combined with CNN for feature extraction further improved the prediction accuracy, with MSE reduced to 15.84 and MAPE reduced to 7.5%. Transformer, as the most advanced time series modeling method, achieved the best performance among all benchmark models, with an MSE of 14.32, an RMSE of 3.78, and a MAPE of only 6.9%. However, its long computation time (5.87 seconds) indicates that Transformer's disadvantage in computational complexity may lead to higher computational costs in practical applications.

The "association rules + LSTM" method proposed in this study outperformed the baseline model in all evaluation indicators, with MSE reduced to 12.51, RMSE only 3.54, and MAPE reduced to 5.8%. Compared with Transformer, it improved the prediction accuracy while optimizing the calculation time to 3.79 seconds, achieving a good balance. This shows that by combining association rules to mine potential patterns in network data and inputting them as features into LSTM for time series prediction, the prediction ability of the model can be effectively improved while reducing the calculation cost, making it more suitable for actual intelligent transportation systems.

4.4 Hyperparameter sensitivity experiments

Next, a hyperparameter sensitivity experiment is given. The experiment mainly focuses on the optimizer and learning rate. First, the experimental results of the learning rate are given, as shown in Table 2.

Table 2: Learning rate experiment results

LR	MSE	RMSE	MAPE
0.005	18.72	4.33	8.5

0.003	15.64	3.95	7.2
0.002	13.87	3.72	6.3
0.001	12.51	3.54	5.8

The experimental results show that the choice of learning rate (LR) has a significant impact on the prediction performance of the model. At a higher learning rate of 0.005, the MSE is 18.72, the RMSE reaches 4.33, and the MAPE is as high as 8.5%, indicating that the model may oscillate during the training process and it is difficult to converge stably, resulting in a large prediction error. This shows that at a larger learning rate, although the model can quickly update parameters, it may not be able to fully capture the temporal characteristics of network flow data, resulting in weak generalization ability. At the same time, a higher learning rate may cause the model to skip the optimal solution area and make it difficult to find the global optimal parameter configuration.

As the learning rate decreases, the prediction error of the model gradually decreases. When the learning rate is reduced to 0.003 and 0.002, the MSE drops to 15.64 and 13.87 respectively, and the MAPE also drops from 7.2% to 6.3%, indicating that the convergence effect of the model at a smaller learning rate is more stable. At this point, the model can learn the changing pattern of network flow more fully and adjust parameters more smoothly during training, reducing the risk of overfitting. However, in the case of 0.002, although the error is further reduced, the convergence speed is relatively slow, and more training iterations may be required to achieve the optimal performance.

When the learning rate is set to 0.001, the experimental results are the best, with MSE reduced to 12.51, RMSE to 3.54, and MAPE to only 5.8%, indicating that the model can be stably trained at this learning rate and effectively capture the temporal characteristics of network flow. A lower learning rate helps the model to gradually optimize parameters during training, so that the loss function converges to a better solution, while avoiding unstable convergence problems caused by excessive gradient step size. Therefore, this study finally selected 0.001 as the optimal learning rate to ensure the stability and high prediction accuracy of the model while maintaining a reasonable computational overhead.

Secondly, the experimental results of different optimizers are given. The experimental results are shown in Table 3.

Table 3: Optimizer experiment results

Optimizer	MSE	RMSE	MAPE
Adagrad	18.95	4.35	8.7
SGD	16.72	4.09	7.9
AdamW	13.84	3.72	6.5
Adam	12.51	3.54	5.8

The Table 3 results show that there are obvious differences in the performance of different optimizers in the network flow prediction task. Adagrad has certain advantages in processing sparse data due to its adaptive learning rate, but it is prone to the problem of rapid learning rate decay in time series tasks, which makes the model unable to continuously optimize. The final MSE is 18.95, RMSE is 4.35, and MAPE is as high as 8.7%. This shows that Adagrad is difficult to fully learn the time series characteristics of network flow in this task, the prediction error is large, and the convergence effect is not ideal. Similarly, SGD, as the most basic optimization algorithm, has high computational efficiency, but due to its simple gradient update method, it is easy to fall into local optimality, resulting in slow model convergence. The final MSE dropped to 16.72, RMSE was 4.09, and MAPE was still high at 7.9%. This shows that there is still a lot of room for optimization in the performance of SGD when processing complex time series data.

In contrast, AdamW combines Adam's adaptive learning rate strategy and introduces L2 regularization during the weight update process, thereby improving the convergence stability of the model. The experimental results show that AdamW's MSE is reduced to 13.84, RMSE is 3.72, and MAPE is reduced to 6.5%, indicating that the optimizer effectively reduces the prediction error while ensuring a faster convergence speed. In addition, AdamW can maintain a relatively stable learning rate during long-term training and will not decay prematurely like Adagrad, which enables the model to learn the long-term pattern of network flow changes more deeply. However, in terms of computational efficiency, AdamW does not have a significant advantage over Adam, so it is necessary to further weigh the computational cost and model performance.

In the end, the Adam optimizer performed best in this experiment, with its MSE reduced to 12.51, RMSE of only 3.54, and MAPE of only 5.8% as well. This shows that Adam can adjust the learning rate more stably when optimizing deep learning models, making the gradient update smoother, thereby improving the convergence and prediction accuracy of the model. In addition, Adam has good robustness when processing nonlinear and complex time series data, and can more effectively capture the time-varying characteristics of network flow. Therefore, considering the comprehensive computational efficiency and prediction accuracy, Adam was finally selected as the optimizer for this study to ensure that the model can achieve the best prediction performance while achieving stable convergence.

4.5 Ablation experiment

In order to further verify the effectiveness of each key module in the model proposed in this study, we designed an ablation experiment to gradually remove different components and evaluate their impact on the final prediction performance. Specifically, the core of the model is time series modeling by LSTM, combined with association rule mining to extract features. Therefore, this experiment examines the use of only LSTM, without adding association rule features, and the impact of different feature selection strategies on model performance. The experiment uses the same dataset, hyperparameter settings, and training environment to ensure the fairness of the comparison, and is evaluated through indicators such as mean square error (MSE), root mean square error (RMSE), and mean absolute percentage error (MAPE). The results of the ablation experiment can reveal the contribution of each module to the overall prediction performance and provide theoretical support for the optimization of the model.

Table 4: Ablation experiment results

Optimizer	MSE	RMSE	MAPE
Full Model (Ours: LSTM + Association Rules)	12.51	3.54	5.8
Without Association Rules	15.23	3.90	7.1
Without LSTM (Only Association Rules + Linear Regression)	18.46	4.30	8.5
Without Feature Engineering (Raw Data to LSTM)	16.78	4.12	7.8

In the case of (100% association rules), the MSE is 12.51, the RMSE is 3.54, and the MAPE is only 5.8%, which is the best performance. This shows that combining LSTM for time series modeling and association rule mining features can effectively improve the accuracy of network flow prediction. When the association rules are removed and only LSTM is used for prediction, the MSE rises to 15.23, the RMSE increases to 3.90, and the MAPE increases to 7.1%, which shows that association rules play an important role in mining network patterns and providing prior knowledge. Models without association rules can only rely on LSTM for end-to-end learning, but lack potential causal information in the data, so the prediction accuracy decreases.

Further removing LSTM and using only association rules and linear regression for prediction, the MSE significantly increases to 18.46, the RMSE increases to 4.30, and the MAPE also increases to 8.5%, indicating that LSTM plays a vital role in time series modeling. Although association rules can provide static relationships of network patterns, they cannot capture complex time dependencies, and the linear regression model has weak expressive power and cannot adapt to nonlinear and dynamically changing network flows, so the overall prediction effect is greatly reduced. In addition, when feature engineering is removed and the original data is directly input into LSTM training, the MSE is 16.78 and the MAPE is 7.8%, which is significantly higher than the error of the complete model, indicating that feature engineering (such as data normalization, time window sliding, and associated feature extraction) plays an important role in improving the learning efficiency and stability of the model.

Overall, the complete model (LSTM + association rules) performs best in all experimental indicators, proving that association rules can effectively improve the prediction ability of LSTM, and feature engineering also makes an important contribution to the optimization of model performance. As a deep learning model, LSTM can fully learn time series features, while association rules supplement implicit pattern information, so that the model can not only predict future trends based on historical data but also combine the inherent correlation in network data, thereby further optimizing the prediction accuracy.

5. Conclusion

This study proposes a network congestion prediction method based on association rules and LSTM, aiming to fully explore the temporal dependency and potential patterns of network traffic data and improve prediction accuracy. Through experimental verification, we found that association rules can effectively extract the correlation between different network factors and provide LSTM with richer input features so that the model has stronger explanatory power when predicting network traffic. At the same time, LSTM, as a powerful time series modeling tool, can capture the long-term change trend of network traffic. When combined with association rules, the prediction performance of the model is significantly improved. Experimental results show that this method outperforms traditional statistical methods, pure LSTM models, and other deep learning benchmark models in multiple evaluation indicators (such as MSE, RMSE, and MAPE), proving the application value of this research method in the field of Internet of Everything.

In the ablation experiment, we further analyzed the contribution of different modules to the model performance. The results show that the synergy of association rules and LSTM is the key factor in improving the performance of the model, and the optimization of feature engineering also significantly affects the final prediction accuracy. When the association rules or feature engineering are removed, the model error increases significantly, indicating that these modules play an important role in learning the complex change pattern of network traffic. In addition, we also studied the effects of different optimizers and learning rates and found that the Adam optimizer and a smaller learning rate (0.001) help improve the stability and convergence speed of the model, thereby further optimizing the prediction effect. These experimental results not only verify the effectiveness of the research method but also provide a reference for the design of future network prediction models.

Although this study has achieved certain results, there are still some issues that deserve further study. First, the current method relies on historical data and association rules for modeling. In the future, more external data, such as accident information, social media data, etc., can be combined to improve the comprehensiveness and real-time nature of the prediction. Secondly, this study mainly uses LSTM for time series modeling. In the future, more advanced time series prediction methods, such as Transformer or hybrid deep learning models, can be explored to further improve the prediction ability of the model. In addition, the existing methods mainly focus on the network traffic prediction of a single road section. In the future, they can be extended to the prediction of multiple road sections or the entire network to improve the applicability of the model in complex network environments. Future research can further explore the application of federated learning and reinforcement learning in network prediction to improve the privacy protection and adaptive optimization capabilities of the model.

References

[1] Yu S, Peng J, Ge Y, et al. A traffic state prediction method based on spatial–temporal data mining of floating car data by using autoformer architecture[J]. Computer-Aided Civil and Infrastructure Engineering, 2024.

[2] Wang L, Ning Z, Cheng Y, et al. Data mining of encrypted network traffic for adult content and gambling Android applications[C]//Third International Conference on Electronic Information Engineering, Big Data, and Computer Technology (EIBDCT 2024). SPIE, 2024, 13181: 1359-1363.

[3] Zhang W, Lei X. Network traffic anomaly detection based on deep learning: a review[J]. International Journal of Computational Science and Engineering, 2024, 27(3): 249-257.

[4] Zheng M, Ma K, Wang F, et al. Which standard classification algorithm has more stable performance for imbalanced network traffic data?[J]. Soft Computing, 2024, 28(1): 217-234.

[5] Peng J, Guo C, Ping Y, et al. SNDMI: Spyware network traffic detection method based on inducement operations[J]. Computers & Security, 2024, 140: 103806.

[6] Rzayev B, Uvaliyeva I, Beldeubayeva Z. The Combined Method for Detecting Anomalies in the Enterprise Telecommunication Networks[C]//2024 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA). IEEE, 2024: 1-4.

[7] Wang X, Wei W, Liu C, et al. Network Traffic Classification with Small-Scale Datasets Using Ensemble Learning[C]//ICC 2024-IEEE International Conference on Communications. IEEE, 2024: 1-6.

[8] Mahalakshmi V, John R, Samuel B E, et al. Exchange matching algorithm for low-complexity traffic scheduling for 5G fronthaul networks[M]//Networks Attack Detection on 5G Networks using Data Mining Techniques. CRC Press, 2024: 127-142.

[9] Luo X, Zhu C, Zhang D, et al. DSTAN: attention-enhanced dynamic spatial-temporal network for traffic forecasting[J]. World Wide Web, 2025, 28(1): 15.

[10] Jeremiah S R, Chen H, Gritzalis S, et al. Leveraging application permissions and network traffic attributes for Android ransomware detection[J]. Journal of Network and Computer Applications, 2024, 230: 103950.

[11] Yang X, Angkawisittpan N, Feng X. Analysis of an enhanced random forest algorithm for identifying encrypted network traffic[J]. EUREKA: Physics and Engineering, 2024 (5): 201-212.

[12] Yun T, Lee K, Yun S, et al. An Offline Meta Black-box Optimization Framework for Adaptive Design of Urban Traffic Light Management Systems[C]//Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. 2024: 6202-6213.

[13] Jiang Q. Communication network security situation analysis based on time series data mining technology[J]. Open Computer Science, 2024, 14(1): 20230104.

[14] Jiang H, Wei Y, Mei L, et al. Multiscale Characteristics and Connection Mechanisms of Attraction Networks: A Trajectory Data Mining Approach Leveraging Geotagged Data[J]. Chinese Geographical Science, 2024: 1-15.

[15] Umamaheswari N, Kumutha K, Kalaivani M S, et al. Backpropagation Shuffled Leaping Neural Network Implementation in Classification of DDoS Packet Flow Traffic in Data Mining[J]. SN Computer Science, 2024, 5(8): 1-7.

[16] Luo J. Traffic Analysis and Optimal Control Technology in Computer Network[C]//2024 International Conference on Power, Electrical Engineering, Electronics and Control (PEEEC). IEEE, 2024: 803-808.

[17] Zuo C, Zhang X, Zhao G, et al. PCR: A Parallel Convolution Residual Network for Traffic Flow Prediction[J]. IEEE Transactions on Emerging Topics in Computational Intelligence, 2025.

[18] Bleijerveld B. A Scalable Recommendation System for Selective Retention of DDoS Traffic: Entropy-Based Network Traffic Analysis[D]. University of Twente, 2024.