electronics

MDPI

Article

A Dynamic Spatio-Temporal Traffic Prediction Model Applicable to Low Earth Orbit Satellite Constellations

Kexuan Liu, Yasheng Zhang * and Shan Lu

Satellite Communications and Broadcasting Department, 54th Research Institute of China Electronics Technology Group Corporation, Shijiazhuang 050081, China; 18131790215@163.com (K.L.); 17732872762@163.com (S.L.)
* Correspondence: zys@163.com

Abstract: Low Earth Orbit (LEO) constellations support the transmission of various communication services and have been widely applied in fields such as global Internet access, the Internet of Things, remote sensing monitoring, and emergency communication. With the surge in traffic volume, the quality of user services has faced unprecedented challenges. Achieving accurate low Earth orbit constellation network traffic prediction can optimize resource allocation, enhance the performance of LEO constellation networks, reduce unnecessary costs in operation management, and enable the system to adapt to the development of future services. Ground networks often adopt methods such as machine learning (support vector machine, SVM) or deep learning (convolutional neural network, CNN; generative adversarial network, GAN) to predict future short- and long-term traffic information, aiming to optimize network performance and ensure service quality. However, these methods lack an understanding of the high-dynamics of LEO satellites and are not applicable to LEO constellations. Therefore, designing an intelligent traffic prediction model that can accurately predict multi-service scenarios in LEO constellations remains an unsolved challenge. In this paper, in light of the characteristics of high-dynamics and the high-frequency data streams of LEO constellation traffic, the authors propose a DST-LEO satellite-traffic prediction model (a dynamic spatio-temporal low Earth orbit satellite traffic prediction model). This model captures the implicit features among satellite nodes through multiple attention mechanism modules and processes the traffic volume and traffic connection/disconnection data of inter-satellite links via a multi-source data separation and fusion strategy, respectively. After splicing and fusing at a specific scale, the model performs prediction through the attention mechanism. The model proposed by the authors achieved a short-term prediction RMSE of 0.0028 and an MAE of 0.0018 on the Abilene dataset. For long-term prediction on the Abilene dataset, the RMSE was 0.0054 and the MAE was 0.0039. The RMSE of the short-term prediction on the dataset simulated by the internal low Earth orbit constellation business simulation system was 0.0034, and the MAE was 0.0026. For the long-term prediction, the RMSE reached 0.0029 and the MAE reached 0.0022. Compared with other time series prediction models, it decreased by 22.3% in terms of the mean squared error and 18.0% in terms of the mean absolute error. The authors validated the functions of each module within the model through ablation experiments and further analyzed the effectiveness of this model in the task of LEO constellation network traffic prediction.

Keywords: low earth orbit constellations; traffic prediction; high dynamics; data separation and fusion; spatio-temporal attention

check for updates

Academic Editor: Paul Mitchell

Received: 1 February 2025
Revised: 25 February 2025
Accepted: 6 March 2025
Published: 6 March 2025

Citation: Liu, K.; Zhang, Y.; Lu, S. A Dynamic Spatio-Temporal Traffic Prediction Model Applicable to Low Earth Orbit Satellite Constellations. *Electronics* **2025**, *14*, 1052. https://doi.org/10.3390/electronics14051052

1. Introduction

A schematic diagram of the communication services provided by a Low Earth Orbit (LEO) constellation [1] to users is shown in Figure 1. Terminals allow users to access satellite Internet services. The gateway station is responsible for the communication connection between satellites and the ground network. The ground control center ensures the normal operation of satellites and the quality of services. The services supported by this system generate a large amount of high-frequency data streams [2]. For example, in dynamic and dense environments (such as cities or disaster-stricken areas), user traffic fluctuates significantly, resulting in high-frequency traffic volatility. Regarding real-time data streams in applications like telemedicine, excessive delays in image transmission may affect the timeliness of diagnosis. In autonomous driving, data delays may impact the real-time control and safety. The LEO constellation can fulfill the task of providing crucial traffic transmission services with excellence, ensuring the timely delivery and processing of information.

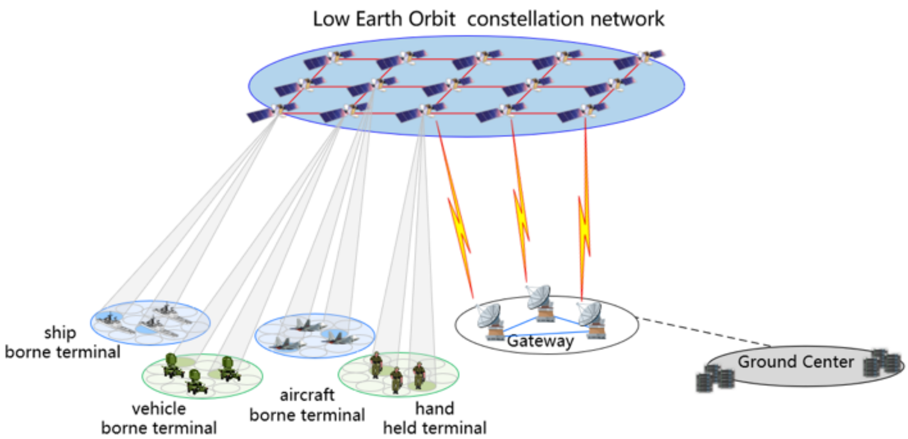

Figure 1. Schematic diagram of the communication services provided to users by LEO constellations.

In order to achieve more efficient network management and optimal resource allocation, traffic prediction is of utmost importance. Traffic prediction for low Earth orbit (LEO) satellites is a crucial technology for maintaining network security and ensuring service quality that aims to predict the link traffic volume in future short- and long-term periods. Traditional methods, such as random forest, comprehensively predict the future traffic volume by integrating features from multiple decision trees including time-related features, basic traffic features, network device and topology features, and user behavior features. However, with the surge in traffic volume, the traffic in LEO constellation links is restricted by factors such as fault maintenance, bandwidth resources, and weather conditions. The inability to accurately predict may lead to irrational resource allocation [3], a decline in service quality, and an increase in operating costs. This poses challenges to ensuring the service quality of the services [4].

In the context of real-world low Earth orbit (LEO) constellation network traffic prediction, the impact of external factors on data transmission is crucial. Atmospheric conditions, electromagnetic interference, satellite equipment performance, inter-satellite link status, ground stations, and various other factors can all affect the communication quality to varying degrees. In terms of atmospheric conditions, for example, during periods of high solar activity, changes in the electron density in the ionosphere can affect signal propagation. Adverse weather conditions such as rain, snow, and fog can absorb and scatter signals, leading to signal attenuation. Regarding electromagnetic interference, solar flares can release large amounts of electromagnetic radiation, disrupting satellite communications. During wartime, human-induced interference from ground-based radar, microwave communication devices, and other equipment can generate electromagnetic interference. The performance of the satellite equipment itself is also critical. Factors such as antenna gain, transmission power, and receiver sensitivity all impact normal communication. In terms of ground stations, the location of a ground station directly affects the visibility of satellites and the duration of communication while the network topology between ground stations and data centers influences the latency and reliability of data transmission. These external factors also present significant challenges in achieving accurate traffic prediction for LEO constellation networks.

To solve the problem of accurately predicting traffic in future time periods, research on network traffic prediction technology has changed significantly over time. In the early stage, some studies [5] achieved network traffic prediction by optimizing prediction models using simulation results through changing the learning rate of the network and the number of hidden-layer units. However, manually-designed networks are overly dependent on high-quality and large-volume data. Some studies [6] used the LightBGM multi-classification algorithm to classify network traffic and then conducted data processing and prediction research on the classified self-similar traffic with strong burstiness. By studying the influence of different parameter values on the prediction accuracy, the optimal traffic prediction model was obtained. With the development of deep learning, one study [7] proposed a method based on spatio-temporal topological graphs. This method considered the correlation between missing data and the available spatio-temporal data as well as the correlation between road network topology and traffic flow, achieving real-time and effective network traffic prediction. Nevertheless, this method is sensitive to data deviation. When considering the magnitude of traffic data, model training is prone to over-fitting. Given the changing dynamic spatio-temporal relationships among satellite nodes, more complex models should be introduced, taking into account the topological relationships and both past and future traffic conditions to achieve accurate network traffic prediction. The network structure of the proposed model is more complex, with a strong capability to learn from large-scale data. By introducing more sophisticated model architectures and regularization techniques, the model significantly reduces the sensitivity to data bias, avoiding prediction errors caused by data bias in traditional methods. The model integrates topological relationships and historical traffic data, employing multi-level attention mechanisms and dynamic weight adjustments, thus effectively preventing overfitting issues during training due to data scale or complexity.

In recent years, the recurrent neural network (RNN) model has been widely used in time-series prediction [8]. Its architecture allows for information to circulate in the network, effectively capturing the temporal dependencies in sequential data. However, when attempting to train deep or long-time-span RNNs, phenomena such as gradient vanishing or explosion may occur, making it difficult to learn the influence of events from the distant past. Moreover, even with careful design and parameter tuning, the standard form of RNNs may still be insufficient to fully represent the correlations between historical patterns over long periods. This is particularly challenging for satellite network traffic that requires the consideration of periodicity. In the face of these challenges, the long short-term memory (LSTM) model emerged. LSTM solves the long-term dependency problem of RNNs by introducing a gating mechanism [9]. Due to its excellent ability to process sequential data, good stability, convergence, and strong robustness, it can effectively process and analyze network traffic data, achieving the accurate prediction of short- and long-term time-series problems. However, LSTM has been criticized by the academic and industrial circles because of its high computational complexity and the risk of over-fitting when the data volume is insufficient. The emergence of the gated recurrent unit (GRU) model [10] has, to some extent, perfectly solved some of the problems of LSTM. GRU simplifies the structure of LSTM and reduces the computational complexity by merging the forget gate and the input gate into an update gate and integrating the cell state and the hidden state. In many sequential data processing tasks, GRU can achieve performances comparable to that of LSTM. Especially in tasks with strong short-term dependencies, GRU may even perform better.

However, in real-world network traffic prediction scenarios, traditional models only focus on the changing trends of traffic volume while neglecting a crucial point: the data in network traffic does not exist in isolation, and there are complex topological relationships behind it. From the connection modes of network nodes to the data interactions between different links, these topological structures have a profound impact on the distribution and variation rules of traffic. Therefore, some research has achieved significant progress in this field by combining the graph convolutional network (GCN) [11] with time-series prediction models. The unique advantage of the GCN lies in its ability to incorporate these topological relationships into the model learning process. Through convolutional operations on graph-structured data, GCN can effectively explore the dependency relationships and feature propagation patterns among nodes in the network, enabling the model to learn the internal connections between traffic data from a more comprehensive perspective. As a result, it demonstrates a more excellent performance in network traffic prediction tasks, significantly improving the accuracy and reliability of prediction. The model proposed by the authors selects LSTM to handle the prediction of extremely large data volumes under surging business demands. Compared with using LSTM alone for network traffic prediction, the authors incorporated a GCN module as a spatial dependency extraction model. The introduction of GCN enables the model to fully exploit the complex relationships within the network topology, integrating the dependencies between nodes and traffic propagation patterns into the learning process. This allows for a more comprehensive understanding of the distribution patterns of traffic data. The combination not only enhances the model's ability to predict traffic changes, but also significantly improves its adaptability to bursty traffic and complex network environments, providing a more accurate and reliable solution for low Earth orbit (LEO) satellite constellation network traffic prediction.

Communication between Low Earth Orbit (LEO) constellations is achieved through inter-satellite links, and the topological structure is shown in Figure 2. Satellites from satellite (1,1) to satellite (4,1) are co-orbital satellites, while those from satellite (1,1) to satellite (1,4) are cross-orbital satellites. Generally, as long as the communication equipment of the satellites is working properly and is not severely interfered with by the space environment, the links [12] between co-orbital satellites can usually remain connected, enabling stable data transmission and information exchange. For cross-orbital satellites, when they operate near the orbital intersection points, the distance between them is relatively short. If the communication equipment has the corresponding capabilities, a link can be established for communication at this time. Alternatively, by deploying relay satellites [13], signals can be forwarded between satellites in different orbits, realizing indirect communication between cross-orbital satellites and improving the possibility and stability of link connectivity. Due to differences in orbital altitude, inclination, etc., the distance and relative angle between cross-orbital satellites change significantly. At certain moments, the link may fail to be established or may be disconnected because the distance is too far or the angle is unfavorable, exceeding the coverage and capabilities of the communication equipment. Therefore, the time-series prediction model incorporating GCN is highly suitable for LEO constellation scenarios. Due to the special characteristics of satellite networks, including high dynamics, complex network topologies, and diverse connections between different satellites, these factors allow the GCN to fully leverage its own advantages.

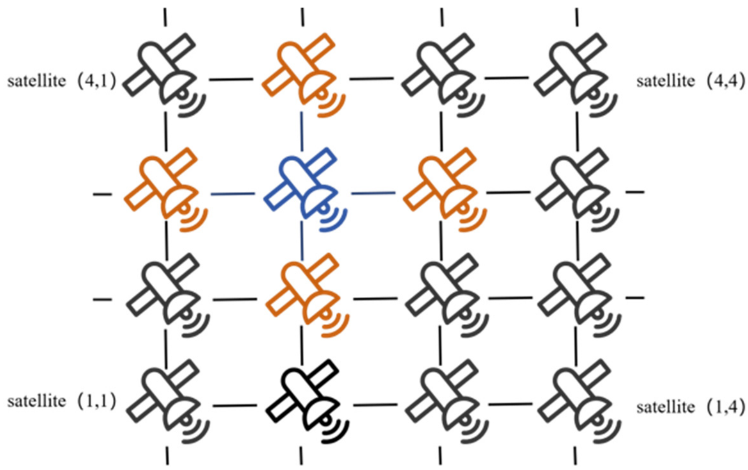

Figure 2. Topological relationship of LEO constellations.

However, most of the current research on the combination of GCN and time-series models has been applied to terrestrial networks such as traffic flow or terrestrial mobile traffic. Although there are topological relationships in terrestrial networks, they are different from those between satellites. The topological structure in LEO constellations changes over time. Currently, there is no modification of the model structure specifically for the task of predicting on-board traffic in LEO satellites. If the method of combining a GCN with a time-series model is directly used to train the model, it will be unreasonable and lack interpretability. The model proposed by the authors also employs data fusion and separation techniques to independently process the collected traffic volume and link the connectivity data. By separately analyzing and then integrating the traffic data and link status data, the model can more accurately capture the spatio-temporal characteristics of network traffic. Additionally, by incorporating a spatio-temporal attention mechanism, the model dynamically adjusts the importance weights of different time steps and spatial nodes, thereby better understanding the dynamically changing topology and traffic distribution patterns in low Earth orbit (LEO) constellations. This design not only improves the model's accuracy in predicting traffic changes, but also enhances its adaptability to bursty traffic and complex network environments.

In this paper, in light of the characteristics of the dynamic topology and high-frequency data streams of LEO constellation network traffic, the authors proposed a DST-LEO (dynamic spatio-temporal low Earth orbit) satellite traffic prediction model to accomplish the task of network traffic prediction in LEO constellations. Specifically, on the basis of the original GCN-time-series prediction model, the model adds multiple attention mechanisms including the traditional attention mechanism and the spatio-temporal attention mechanism. These mechanisms were used to capture the more subtle changes in the satellite positional relationships caused by time variations that were previously ignored. These details directly affect the presence or absence of traffic (for example, the disconnection of cross-orbital satellites). Afterward, a multi-source data separation and fusion strategy [14] was adopted to split the data of traffic volume and traffic connection/disconnection into two parts for processing. This enabled the model to fully learn the characteristics of the data with different physical meanings. Subsequently, the data were spliced and fused in specific dimensions. Finally, the ultimate prediction of the LEO constellation network traffic was achieved through the attention mechanism and the fully-connected layer.

2. Research Status

2.1. Traditional Traffic Prediction Models

Time-series analysis models: Some studies [15] have used the Holt–Winters model to achieve time-series prediction and found that the traffic flow pattern not only involves random patient movements, but also shows trends and seasonal variation rules. Another study [16] proposed a multi-mean fusion prediction model that decomposed the traffic input into three parts: the trend sequence, the up–down sequence, and the noise sequence. Different models were used to fit and predict these sequences, and finally, a linear fusion of the three prediction results was carried out to obtain the changing characteristics of the predicted traffic pattern and improve the accuracy of the time-series prediction. However, satellite network traffic is extremely complex and is accompanied by high-frequency data streams. Although traditional methods perform well in predicting individual future time periods, it is very difficult for them to learn the traffic patterns in scenarios with multiple nodes, high-frequency data, and highly dynamic spatio-temporal topologies. Our model can better capture the potential hidden patterns of LEO constellations through multiple attention mechanisms, enabling accurate traffic prediction.

Machine learning models: With the development of machine learning, this approach has been widely applied in the field of time-series prediction. Sajal Saha et al. [17] developed an ITF model to assist ISPs (Internet service providers) in long-term business strategic planning and proactive network management. This model combines outlier detection and mitigation techniques with advanced gradient descent and boosting algorithms, significantly enhancing the model's performance and enabling real-world Internet traffic monitoring in high-speed ISP networks. Rau Francisco et al. [18] compared the online sequential extreme learning machine (OS-ELM) with other machine learning models and demonstrated that OS-ELM outperformed other networks in terms of both the accuracy and computational efficiency, offering a promising solution for network traffic prediction problems. However, machine learning models typically train on only one type of prediction data. Our method separates the collected LEO constellation traffic into data on traffic volume at different times and data on traffic connection and disconnection through multi-source data separation. Time-series prediction models are then trained separately to learn the characteristics of the data with different physical meanings and combine them through splicing. Together with multiple attention mechanisms and GCN, our method can achieve more accurate network traffic prediction.

2.2. Recurrent Neural Network Models

RNN models and their variants: In the early days, some research [19] combined RNNs and CNNs (convolutional neural networks). The RNN was used to process scalar variables to extract the overall features of the network, while the CNN was used to extract traffic information for dealing with specific network configurations, predicting potential time-series problems, and achieving good air traffic flow management. However, since RNN models are prone to gradient vanishing and gradient explosion, subsequent academic research has mostly used RNN variants such as LSTM and GRU to solve time-series prediction problems. Yu Zeyang et al. [20] studied a long-short-term memory network model applied to named data networking. They expanded the single traffic feature value to multi-dimensional variables related to link traffic, achieving precise control of the network traffic and avoiding network congestion. Jiang Weiwei first modeled the Internet traffic matrix prediction problem as a video prediction task [21]. Then, a Seq2Seq model based on a convolutional long-short-term memory network (ConvLSTM-TM) was proposed to predict the traffic matrix in the next time slot, completing the time-series prediction of network traffic and providing a new idea for the academic community. Ning Li et al. [22]

proposed a gated recurrent unit (GRU) neural network traffic prediction algorithm based on transfer learning and realized the expression of the nonlinear, self-similar, and long-range dependent characteristics of satellite network traffic. By combining the transfer learning method, they solved the problem of insufficient online traffic data. They also adopted a particle filter online training algorithm to reduce the training time complexity and achieve an accurate prediction of the satellite network traffic.

2.3. Attention Mechanism Models

With the widespread adoption of the self-attention mechanism in transformer, researchers have found that incorporating attention mechanisms can help models capture more subtle features of traffic patterns and achieve precise time-series prediction. Some research [23] introduced an attention mechanism to adjust the importance of traffic information at different time points, thus solving the problem of traffic prediction based on static spatio-temporal dependencies in software-defined networking (SDN) traffic engineering. Another study [24] also introduced an attention mechanism and combined it with the GRU neural network. This approach fully exploited the self-similarity and long-range correlation characteristics of traffic data sequences, focused on the importance of traffic data and hidden states, and learned the time-dependent characteristics of the input sequences. In general, accurately grasping the key features and long-term dependencies of data is crucial. The emergence of attention mechanisms has brought new ideas for solving time-series prediction problems. When combined with recurrent neural network models, they complement each other and greatly enhance the performance of time-series prediction models.

3. DST-LEO Satellite-Traffic Prediction Model

3.1. LEO Constellation Traffic Prediction Architecture

In this section, the aim was to enable the DST-LEO satellite-traffic prediction model to learn various communication patterns and potential features of the low Earth orbit (LEO) constellation network traffic and predict the traffic volume of each link in both short-and long-term future periods. Therefore, the authors used two datasets to verify the rationality and superior performance of the model. The authors introduced a spatio-temporal attention mechanism into the model to help it capture the unique patterns and features in inter-satellite links. At the data processing level, the authors introduced data separation, enabling the model to learn more targeted feature representations in the GCN layer and the LSTM layer according to the respective characteristics of the traffic amplitude changes and link connectivity.

The traffic prediction tasks in low Earth orbit (LEO) constellations are quite different from those in terrestrial networks. In terrestrial networks, the graph topology structure introduced by the graph convolutional network (GCN) remains unchanged. However, in inter-satellite scenarios, due to the dynamic topology of satellites, the satellites will change the topology structure as they move. In this case, the GCN needs to be combined with a spatio-temporal attention mechanism to play a greater role. In addition, LEO constellations carry a large number of high-frequency data streams, and the amplitude of traffic changes is relatively large. A simple LSTM model is insufficient to capture the unique patterns and features. Multiple bi-LSTM layers, combined with GCN, specific data processing methods, and the spatio-temporal attention mechanism, can more effectively and deeply explore the laws of traffic information. To achieve accurate traffic prediction, the authors proposed a traffic prediction architecture for LEO constellations, which can be divided into the following parts, as shown in Figure 3: (1) Collect the traffic of inter-satellite links in the bearer network. (2) Process the traffic into a matrix-based dataset for model training. (3) Separate the collected traffic volume data and link the connectivity data through multi-source data separation and input them into the model, respectively. Then, unify the outputs to a specific dimension through data fusion. Enhance the model's understanding of features through the spatio-temporal attention mechanism, and finally, achieve the prediction of LEO constellation network traffic through the fully-connected layer.

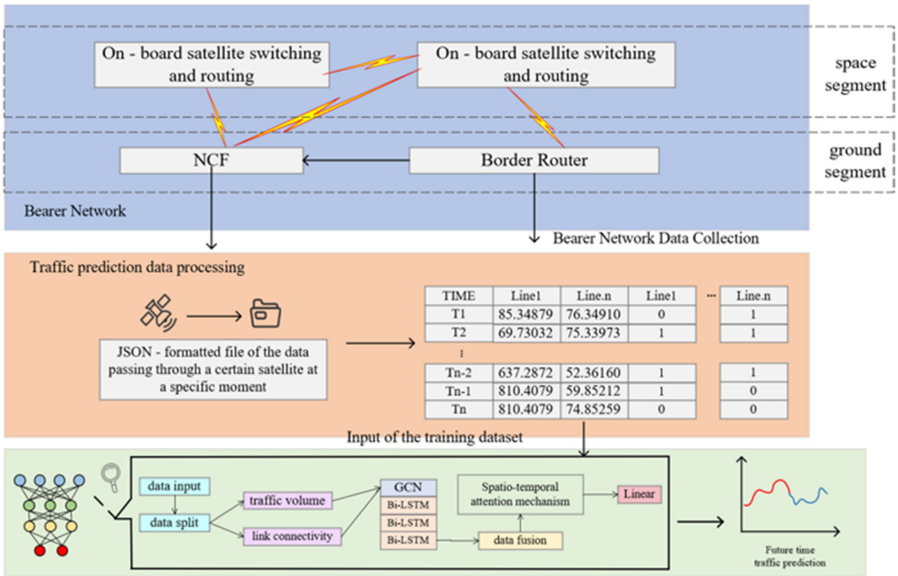

Figure 3. Flowchart architecture of the overall traffic prediction for LEO constellations.

3.2. Traffic Collection of Satellite Bearer Networks

To collect traffic data of low Earth orbit (LEO) constellations from the bearer network, the following steps are generally involved. Firstly, it is necessary to determine the types of traffic data. These include traffic volume, traffic direction, protocol types (such as TCP and UDP), source and destination IP addresses, etc. Secondly, the traffic collection points need to be identified. The network control function (NCF) is responsible for network control and management and can obtain global network status information. The border router, as a network boundary device, is in charge of communicating with other networks (such as ground stations or other satellite networks) and can capture the traffic data flowing in and out of the network. The simple network management protocol (SNMP) is used to collect network status information from the NCF such as traffic volume statistics and link connection/disconnection status. The port mirroring function of the border router is configured to copy the traffic to the monitoring port, and then a packet-capturing tool is used to capture the data. Finally, the collected data are stored in a database as structured data, which facilitates subsequent data processing.

3.3. Traffic Data Processing

Since the authors only focused on the magnitude of the traffic, which in turn helps to better analyze issues such as network resource optimization, capacity planning and expansion, and the quality of service assurance in subsequent research, the authors did not consider the traffic flow direction. Instead, the authors only selected the magnitudes of the data flowing in and out at a certain moment on a single link and added them together. At the same time, the connection and disconnection status of the link at the same moment was extracted. The processed data were provided for model training. It is necessary to ensure that the magnitudes of the traffic and the connection and disconnection status of the links are the same in quantity and correspond one-to-one. Figure 4 shows the processed data format of two inter-satellite links among them. The data included the magnitudes of the traffic on the two links within the corresponding time period as well as the connection and disconnection status of the links. 0 means that the inter-satellite link is disconnected, and 1 represents that the link is in normal communication. The dataset was constructed according to this format, and the magnitudes of the traffic in the first N columns need to correspond to the connection and disconnection status of the links in the subsequent N columns.

	A	B	C	D	E
1	DATETIMESTAMP	line1 (volume)	line2 (volume)	line1 (connection)	line2 (connection)
2	2023/3/17 15:54	85.34879	76.34910	1	1
3	2023/3/17 15:59	85.34879	76.34910	1	0
4	2023/3/17 16:04	85.34879	76.34910	1	1
5	2023/3/17 16:09	85.34879	76.34910	0	1
6	2023/3/17 16:14	85.34879	76.34910	1	1
7	2023/3/17 16:19	85.34879	76.34910	1	1
8	2023/3/17 16:24	85.34879	76.34910	1	1
9	2023/3/17 16:29	85.34879	76.34910	1	1
10	2023/3/17 16:34	85.34879	75.33973	1	0
11	2023/3/17 16:39	85.34879	75.33973	1	1

Figure 4. Schematic diagram of the data format for training.

3.4. Model Architecture

The main architecture of the DST-LEO satellite-traffic prediction model consists of a data separation module, a spatial dependency extraction model, a temporal dependency extraction model, a data fusion module, an attention mechanism, and an output layer. Due to the complexity of the dynamic topology, the model utilizes a graph convolutional network (GCN), the data separation and fusion module, and the spatio-temporal attention mechanism. These components complement each other, enabling the model to accurately capture the changing patterns of traffic.

3.4.1. Data Separation and Data Fusion Module

By processing the input in the in_channels dimension, the traffic volume and link connectivity are split into tensors of shape (batch_size, num_nodes, in_channels//2). Multi-source data processing can simplify the difficulty of data handling. Data from different types of satellites usually have different characteristics, formats, and noise distributions. This approach avoids interference between different types of data and enhances the data specificity. After separation, it is possible to analyze and mine data from specific sources more precisely and explore the information contained in different data in-depth.

The features of traffic volume and link connectivity are concatenated and fused, resulting in a feature channel number of 2*hidden_channels*2 (representing bidirectionality). This method makes full use of the unique information of different data and also discovers the potential relationships between them. Compared with directly processing mixed data, it can more effectively improve the performance of the model.

3.4.2. Spatial Dependency Extraction Model

The model employs 64 graph convolutional network (GCN) layers to process the traffic volume and link connectivity. The parameters of each GCN layer are as follows: hidden_channels = 64 and activation = "relu". After each GCN layer, there is a BatchNorm1d layer for normalization and a dropout layer to prevent overfitting.

The principle of the graph convolutional network (GCN) is to construct a filter in the Fourier domain and process the graph nodes and their first-order domain through this filter, in order to obtain the spatial features of the nodes in the graph. Finally, the spatial dependency extraction model is established through the superposition of 64 convolutional layers. In this paper, the low Earth orbit (LEO) constellation network traffic model matrix was used as the node features, and 64 convolutional layers were designed to process the graph structure. The formula is as follows:

$$H^{(l+1)} = ReLU\left(\widetilde{D}^{-\frac{1}{2}}\widetilde{A}\widetilde{D}^{-\frac{1}{2}}H^{(l)}W^{(l)}\right), l = 0, 1, 2, \ldots, 63 \tag{1}$$

Here, $H^{(1)}$ is the input feature matrix with the shape of (N, F1), where N represents the number of nodes in the graph. Specifically, for 104 satellite links, N = 104. F1 is the number of input features, which mainly depends on the selection of node features. For example, in this paper, the traffic volume and the link on-off data were selected.. The output feature matrix of the 64th layer is $H^{(64)}$ with the shape of (N, F64), where F64 is the dimension of the output features. $\widetilde{A} = A + I$,$\widetilde{A}$ is the adjacency matrix with self-loops added, where A is the original adjacency matrix and I is the identity matrix. $\widetilde{D} = \sum_j \widetilde{A}_{ij}$,$\widetilde{D}$ is the degree matrix of $\widetilde{A}$. $W^{(l)}$ is the learnable weight matrix of the l-th layer, with a shape of (F_l, F_{l+1}). The rectified linear unit (*ReLU*) activation function is selected, which plays a crucial role. Firstly, by introducing nonlinearity, it enhances the model's ability to learn nonlinear data. Secondly, *ReLU* can alleviate the vanishing gradient problem and accelerate the model's convergence. Finally, *ReLU* enables sparse activation. While reducing redundant computations, it can also automatically select important features. $\widetilde{L} = \widetilde{D}^{-\frac{1}{2}}\widetilde{A}\widetilde{D}^{-\frac{1}{2}}H^{(l)}$ represents the normalized Laplacian matrix, that is, the graph convolution operator.

The BatchNorm normalization technique and residual connections were used to stabilize the training process. The meanings of the symbols are the same as those in Formula (1). Normalization can accelerate the convergence of the model and improve its generalization ability and performance. The formulae are as follows:

$$H^{(l+1)} = Norm\left(ReLU\left(\widetilde{D}^{-\frac{1}{2}}\widetilde{A}\widetilde{D}^{-\frac{1}{2}}H^{(l)}W^{(l)}\right) + H^{(l)}\right), l = 0, 1, 2, \ldots, 63 \tag{2}$$

3.4.3. Temporal Dependency Extraction Model

In this paper, the bidirectional long short-term memory (Bi-LSTM) units were selected to capture the temporal dependencies. The number of layers was set to 3, hidden_channels = 64, and the time step num_lags = 24. The traffic volume and link connectivity were processed through the Bi-LSTM layers, respectively. For a unidirectional LSTM unit, the input X_t is the input at time step t, h_{t-1} is the hidden state at the previous time step, C_{t-1} is the cell state at the previous time step, and the forward-propagation formula is as follows:

Input gate:
$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + b_i) \tag{3}$$

Forget gate:
$$f_t = \sigma\left(W_{xf}x_t + W_{hf}h_{t-1} + b_f\right) \tag{4}$$

Candidate cell state:
$$\widetilde{C}_t = tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \tag{5}$$

Cell state update:
$$C_t = f_t \odot C_{t-1} + i_t \odot \widetilde{C}_t \tag{6}$$

Output gate:
$$O_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + b_o) \tag{7}$$

Hidden state update:
$$h_t = o_t \odot tanh(c_t) \tag{8}$$

Among them, σ is the Sigmoid activation function, ⊙ represents element-wise multiplication,$W_{xi}, W_{hi}, W_{xf}, W_{hf}, W_{xc}, W_{hc}, W_{xo}, W_{ho}$ are trainable parameter vectors, and b_i, b_f, b_c, b_o are the trainable bias vectors.

The Bi-LSTM consists of two independent LSTM units: a forward LSTM and a backward LSTM. The forward input sequence is $x_1, x_2, \ldots, x_t$, and the backward input sequence is $x_t, x_{t-1}, \ldots, x_1$. The hidden state at time step t of the backward LSTM is $\overleftarrow{h_t}$, and the cell state is $\overleftarrow{c_t}$. The output of the Bi-LSTM is the concatenation of the forward and backward hidden states: $h_t = \left[\overrightarrow{h_t}; \overleftarrow{h_t}\right]$, with the shape of (batch_size, 2* hidden_dim).

3.4.4. Attention Mechanism

In this paper, the authors introduced temporal attention and spatial attention to enable the model to learn the importance of traffic information at each moment. This was used to process the low Earth orbit satellite traffic data involving time and space. Its goal is to extract important features by capturing the temporal and spatial dependencies in the data. The design steps of the attention mechanism are as follows:

The input data are x with the shape of (B, T, N, F), where B is the batch_size, T is the length of the time series, N is the number of nodes, and F is the feature dimension. spatial_neighbors is the adjacency matrix.

Temporal attention calculation: For each time lag i (ranging from 1 to num_lags. Here, the setting of the time step needs to consider the actual situation and the data collection frequency. In Section 4.1, as the collection frequency of the Abilene dataset was 5 min, it is reasonable to set the time step to 24 (2 h), and as the collection frequency of the traffic data in the internal low Earth orbit constellation business simulation system is 5 s, it is reasonable to set the time step to 30 (1 min). An overly long time step will lead to high memory usage, introduce noise, and even cause model overfitting, while an overly short time step may limit the learning ability of the model. The historical data and the current data are extracted for the lagged data lagged_x = x[:,:-i,:,:] and the current data current_x = x[:,i:,:,:], both of which have the shape of (B, T-i, N, F). Next, the historical data and the current data are weighted using the temporal attention weights. The formula is as follows, and the resulting data also have the shape of (B, T-i, N, F). ":" indicates the selection of all elements in that dimension. For example, x[:, :, :, :] means selecting all elements of the tensor x. ":i" means taking the elements from the i-th position (including the i-th position) of that dimension to the end of that dimension, while ":-i" means taking the elements from the starting position of that dimension to the i-th position from the end (excluding the i-th position from the end).

$$attention^{(i)} = \sum_{f=1}^{F} W_{temp}^{(i-1)} \cdot lagged\_x_{:,:,:,f} \cdot current\_x_{:,:,:,f} \tag{9}$$

The attention scores of all time lags are stored and concatenated together. The formula is as follows, and the resulting tensor has the shape of (B, T-num_lags, N, num_lags).

$$\text{temporal\_attentions} = \left[\text{attention}^{(1)}, \text{attention}^{(2)}, \cdots, \text{attention}^{(\text{num\_lags})}\right] \tag{10}$$

$$\text{temporal\_attention} = \text{concat}(\text{temporal\_attentions}, \text{dim} = 1) \tag{11}$$

Spatial attention calculation: Expand the data x to the shape of (B, T, N, 1, F) so that it can be multiplied with the spatial adjacency matrix. Weight the input data and the spatial adjacency matrix using the spatial attention weights $\mathrm{W_{spatital}}$. The formula is as follows, and the shape of spatial_attention is (B, T, N, 1).

$$\text{spatial\_attention} = \sum_{\text{f}=1}^{\text{F}} \text{W}_{\text{spatial}}^{(\text{f})} \cdot \text{x}_{:,:,:,\text{f}} \cdot \text{spatial\_neihbors} \tag{12}$$

Expand the weighted spatial_attention to the shape of (B, T-num_lags, N, num_lags), then add it to the temporal attention, and perform a softmax operation to calculate the attention weights.

$$\text{combined\_attention} = \text{temporal\_attention} + \text{spatial\_attention} \tag{13}$$

$$\text{attention\_weights} = \text{softmax}(\text{combined\_attention}, \dim = 1) \tag{14}$$

Expand the input data x to the shape of (B, T-num_lags, N, num_lags, F), and then use the attention weights to perform a weighted sum on the input data. The formula is as follows, and the resulting data has the shape of (B, N, F), representing the weighted features of each node.

$$\text{output} = \sum_{t=1}^{T-\text{num\_lags}} \sum_{i=1}^{\text{num\_lags}} \text{attention\_weights}_{:,t,:,i} \cdot x_{:,t,:,:} \tag{15}$$

4. Result Analysis

4.1. Datasets

This subsection introduces the datasets used for model training in the paper. First, to verify the model's performance, the authors selected the dynamic traffic part of the Abilene dataset from the SNDlib of the terrestrial network. The Abilene data are an important dataset for network research. The Abilene network consists of 12 main nodes, which are the key hubs in the network and undertake functions such as data reception, processing, and forwarding. Data were collected every 5 min daily for six months. The network structure of Abilene is shown in Figure 5. Since the data volumes between four paths (ATLAng and ATLAM5, ATLAng and IPLSng, IPLSng and CHINng, WASHng and NYCMng) were relatively small, only the data sizes of the other 11 paths were processed for model training. The adjacency matrix required by the GCN was set according to the connection relationships in the figure. If there was a path between two nodes, it was regarded as 1 (connected); otherwise, it was regarded as 0 (disconnected). The reasons for selecting this dataset are as follows. Firstly, the Abilene dataset is from a real-world terrestrial network, with data collected from the backbone network in the United States. Secondly, there are obvious topological relationships on the ground. For example, if the network traffic passing through the DNVRng node increases, the traffic of adjacent nodes will also increase accordingly. The model can learn the network topology through the GCN. The topology between satellites at a certain moment is similar, and the GCN can detect the traffic changes between adjacent topological structures.

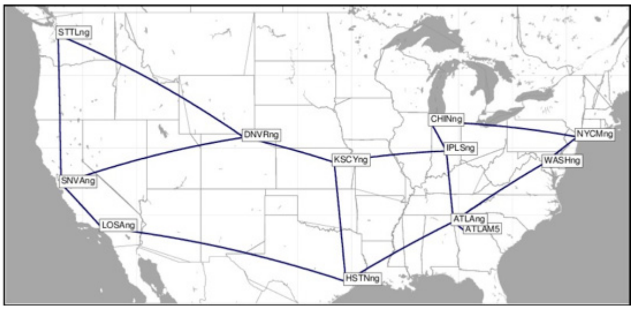

Figure 5. Schematic diagram of the network structure of the Abilene dataset.

Next, after using the ground network traffic dataset experiment to demonstrate the superior performance and shortcomings of the model, further practice was carried out using the low Earth orbit (LEO) constellation verification environment simulated by the

LEO constellation service simulation system developed internally by the research unit where the authors are located to optimize the performance of the model.

This verification environment simulated sixty LEO satellites, one protocol gateway station, and multiple terminals. Each LEO satellite was equipped with an on-board routing and switching node that ran an inter-satellite routing protocol to achieve data forwarding within and between satellites. The specific details are shown in Figure 6. Satellite terminals simulated the network layer and various communication-related functions of real-user terminal devices. Two end-user servers simulated multiple services such as video, voice, text, and pictures, and achieved communication through the simulated satellite system. To better match the real-world changes in system service traffic, the inter-satellite links not only switched on and off according to changes in position, but also simulated the communication impacts caused by external factors such as faults, atmospheric conditions, and electromagnetic interference. The authors designed the external interference factors from two aspects. On the one hand, the authors created a natural-factor pattern that followed a Poisson distribution. In this case, the traffic volume on the link was reduced to 80% of the original to simulate packet loss during service transmission. On the other hand, in the simulation state of the normal system operation, a certain inter-satellite link carrying services was randomly selected to generate a terminal failure, thereby simulating the sudden failure of load-bearing equipment in the operating system. Relevant researchers collected the traffic data of 104 links at a time interval of 5 s. The traffic of a certain satellite node at a certain moment was collected via IP and then summed up, and the on–off status of the inter-satellite links at that moment was recorded. These data were used for subsequent model training.

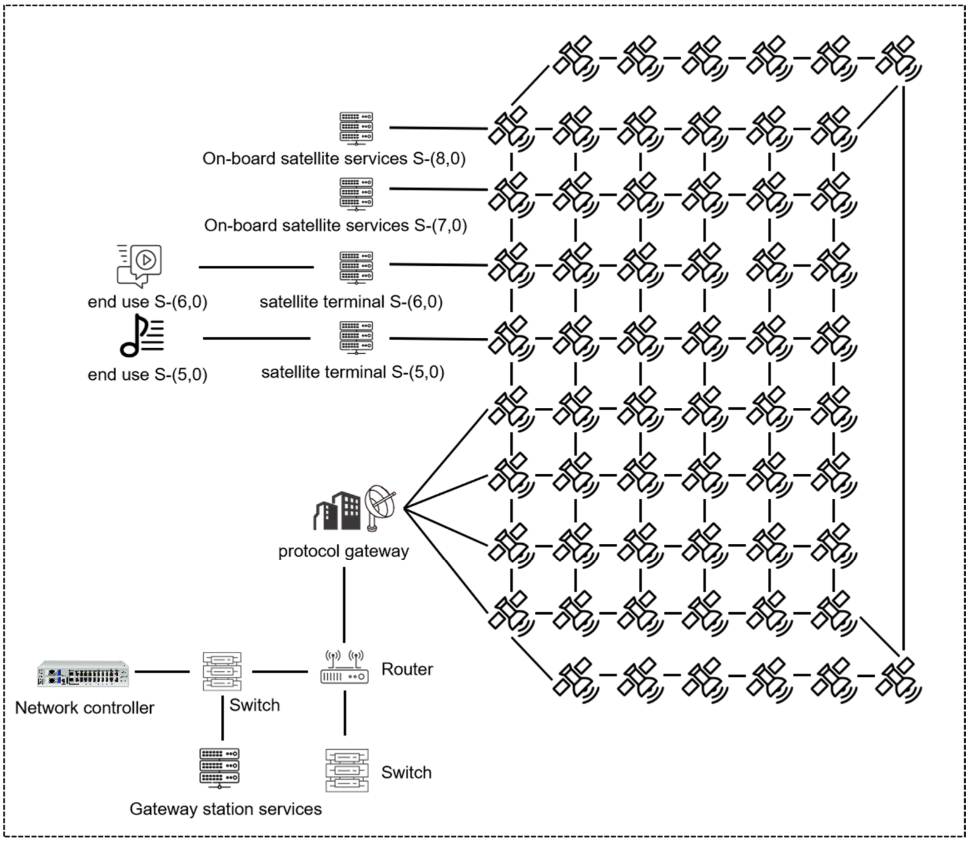

Figure 6. Structural diagram of the internal low earth orbit constellation service simulation system.

4.2. Experimental Environment and Parameter Settings

This experiment was implemented using Python 3.11, Torch 2.2.1, and the relevant Python statistical packages. The data were collected from the internal Low Earth Orbit constellation business simulation system. Both the compilation and testing of the experiment were completed on a hardware device with an AMD Ryzen 7-5800 CPU (AMD, Santa Clara, CA, USA) and an RTX 3070 GPU (NVIDIA, Santa Clara, CA, USA). In the traffic prediction experiment of this paper, the traffic matrix from Section 4.1 was used for single-step prediction, with 80% of the data as the training set and 20% as the test set. The learning rate of the model was determined through preliminary experiments. A learning rate that is too large may lead to unstable training, while one that is too small may result in slow convergence. After multiple attempts, the learning rate was set to 0.027. The batch size needs to balance computational resources and model convergence. For an RTX 3070, setting the batch size to 16 may cause unstable gradient updates, while setting it to 64 would impose significant memory pressure. Therefore, the batch size was set to 32. The node feature dimension was 1, and the hidden feature dimension was 64. This hidden feature dimension provides sufficient expressive power to capture complex patterns in the traffic data without excessively increasing the number of model parameters. The training cycle was 300 epochs, as the loss function tends to stabilize at this point without overfitting. The loss function used was the mean squared error (MSE), and the Adam optimizer was employed for gradient descent training. The Adam optimizer combines the advantages of momentum and adaptive learning rates, automatically adjusting the learning rate for different parameters, making it suitable for handling non-stationary objective functions.

4.3. Experimental Settings

4.3.1. Evaluation Metrics

In this section, the authors conducted two traffic prediction tasks and made predictions for different time lengths, respectively. The authors hoped to predict the traffic changes within the next day (short-term prediction) and the next week (long-term prediction) to cooperate with traffic scheduling, load balancing, and other means to achieve the health monitoring of the low Earth orbit constellation traffic system. By comparing our proposed model with other existing models, the excellent performance of the DST-LEO satellite-traffic prediction model was demonstrated. The evaluation metrics were the mean squared error (MSE), root mean squared error (RMSE), mean absolute error (MAE), and mean absolute percentage error (MAPE), and $X_i, \widetilde{X}_i$ are the predicted values and the true values, respectively.

$$\mathrm{MSE} = \frac{1}{m}\sum_{i=1}^{m}\left(X_i - \widetilde{X}_i\right)^2 \tag{16}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{m}\sum_{i=1}^{m}\left(X_i - \widetilde{X}_i\right)^2} \tag{17}$$

$$\mathrm{MAE} = \frac{1}{m}\sum_{i=1}^{m}\left|\left(X_i - \widetilde{X}_i\right)\right| \tag{18}$$

$$\mathrm{MAPE} = \frac{1}{m}\sum_{i=1}^{m}\left|\frac{X_i - \widetilde{X}_i}{X_i}\right| \tag{19}$$

The four metrics have different characteristics and functions. The mean squared error (MSE) amplifies the impact of large errors through the form of squared errors, so it is quite sensitive to outliers. In a network environment, significant traffic changes caused by sudden situations may have a large impact on the system, and MSE can help the model focus better on such situations. The root mean squared error (RMSE) is the square root of the MSE. It solves the problem of inconsistent dimensions in the MSE, making the result have the same dimension as the original data, which is convenient for observation, understanding, and interpretation. The mean absolute error (MAE) directly calculates the absolute error between the predicted value and the true value. It has a lower sensitivity to outliers and can more robustly reflect the overall performance of the model rather than just focusing on the prediction at a certain moment. The mean absolute percentage error (MAPE) represents the error in percentage form, which can intuitively reflect the proportion of the prediction error relative to the true value, facilitating cross-dataset comparison. However, when the true value is close to zero, the MAPE becomes unstable. Regardless of whether it is MSE, RMSE, MAE, or MAPE, the smaller their values, the higher the prediction accuracy of the model. By combining multiple metrics, the performance of the model can be evaluated more comprehensively.

This paper did not evaluate the model in terms of the time complexity of training. Firstly, due to the fact that the dimensionality of this dataset was not large, the model's training time was insufficient to serve as an effective measurement indicator. Secondly, the four models had different numbers of training rounds. ARIMA only required 100 training rounds to achieve a good fitting effect, and the loss value no longer decreased. In contrast, the other models required 285 or 300 training rounds. Additionally, when conducting long-term prediction on the Abilene dataset, the LSTM-teacher forcing model required approximately 600 training rounds. Although the time complexity of training is also important, this paper paid more attention to the model's ability to capture traffic patterns to better predict future traffic.

4.3.2. Comparison with Existing Models

ARIMA: ARIMA (autoregressive integrated moving average) [25] is a traditional time-series analysis method that is mainly used for modeling and predicting stationary time series. This model has strong capabilities in modeling linear relationships and can capture trends and time-period characteristics in a time series (for example, the traffic reaches a peak during the busy communication business hours in a day). Moreover, the model structure is relatively simple and highly interpretable. However, its ability to model complex nonlinear relationships is limited. By combining it with neural network models, its ability to model nonlinear relationships can be enhanced, enabling it to more comprehensively capture the features and patterns in time-series data and improve the accuracy of predictions.

BiGRU-Attention [26]: Due to its simple structure, the gated recurrent unit (GRU) reduces computational complexity. When the amount of training data is small, GRU is less prone to overfitting than the long short-term memory (LSTM) network. Moreover, when storing model parameters and intermediate calculation results, GRU requires less memory space than LSTM. This model introduces a simple attention mechanism that mainly focuses on the information weighting in the feature dimension. It treats the input features as a whole. By calculating the importance weights of each feature vector, it performs a weighted sum on the feature vectors to highlight more important feature information. This mechanism does not take into account the temporal and spatial dimension characteristics of the data, and simply distributes attention from the perspective of features.

LSTM-Teacher Forcing: Teacher forcing [27] is a training technique used when training recurrent neural networks (RNNs) and their variants (such as LSTM and GRU) for sequence generation tasks. In the traditional training method, the output of the model at the previous time step is used as the input for the next time step. However, this approach may lead to error accumulation. Once the model makes an incorrect prediction at a certain time step, subsequent predictions may be severely affected, making it difficult for the model to converge to good results. In the teacher-forcing mechanism, during the training process, with a certain probability, elements from the true target sequence are directly used as the input for the next time step, rather than using the model's own prediction result from the previous time step. This allows the model to learn the correct sequence patterns more quickly and speeds up the convergence process.

The DST-LEO satellite-traffic prediction model demonstrated its advantages in handling low Earth orbit (LEO) constellation traffic prediction through a comparison with the aforementioned existing models. Although ARIMA is strong in handling the linear relationships in data, the traffic in LEO constellations is intricate, and inter-satellite links are affected by various factors. The comparison proved the advantages of recurrent neural network-like models. Since the LEO constellation supports an extremely large volume of services and the traffic amplitude changes significantly, a deviation in the prediction result at a certain moment will affect subsequent predictions. Therefore, the LSTM-Teacher Forcing mechanism provided the authors with new ideas for future research. The BiGRU-Attention model adopts a simple attention mechanism, while the authors' model introduced a spatio-temporal attention mechanism to separately process the traffic volume and link on-off data, aiming to capture the spatio-temporal dependencies and features in the data. The comparison proved the rationality and advantages of the spatio-temporal attention mechanism in the task of LEO constellation traffic prediction. Moreover, due to the large daily data volume of the constellation, LSTM has a more complex structure to learn the implicit patterns of traffic services, which is also one of the reasons why the authors chose LSTM over GRU.

4.4. Experimental Results

The authors predicted the traffic change trend for the next day through experiments on the Abilene dataset and compared the performance of the four models. The evaluation metrics are shown in Table 1, and the prediction curve for Link 1 is shown in Figure 7. Since ARIMA failed to learn the hidden patterns among nonlinear traffic, the model could only predict the general trend of traffic changes, and there were significant differences in the predicted traffic volume. The prediction result of the BiGRU model with an attention mechanism barely met the authors' expectations, but was not precise enough, possibly due to its simple model structure. Both the LSTM with the teacher-forcing mechanism and the DST-LEO satellite-traffic prediction model achieved the best results in terms of both the predicted trends and peaks, which greatly met the authors' expectations. However, the DST-LEO satellite-traffic prediction model outperformed the others in terms of detail processing, with a MAPE of 2.1315 and an MAE of 0.0018.

Table 1. Evaluation metrics for the short-term prediction of the Abilene dataset.

Dataset	**Abilene**			
Models	**MSE**	**RMSE**	**MAE**	**MAPE**
ARIMA [25]	0.0352	0.1875	0.1583	144.0793
BiGRU-Attention [26]	0.0002	0.0157	0.0096	7.0319
LSTM-Teacher Forcing [27]	3.92×10^{-5}	0.0063	0.0045	4.2500
DLS-Traffic Prediction	7.67×10^{-6}	0.0028	0.0018	2.1315

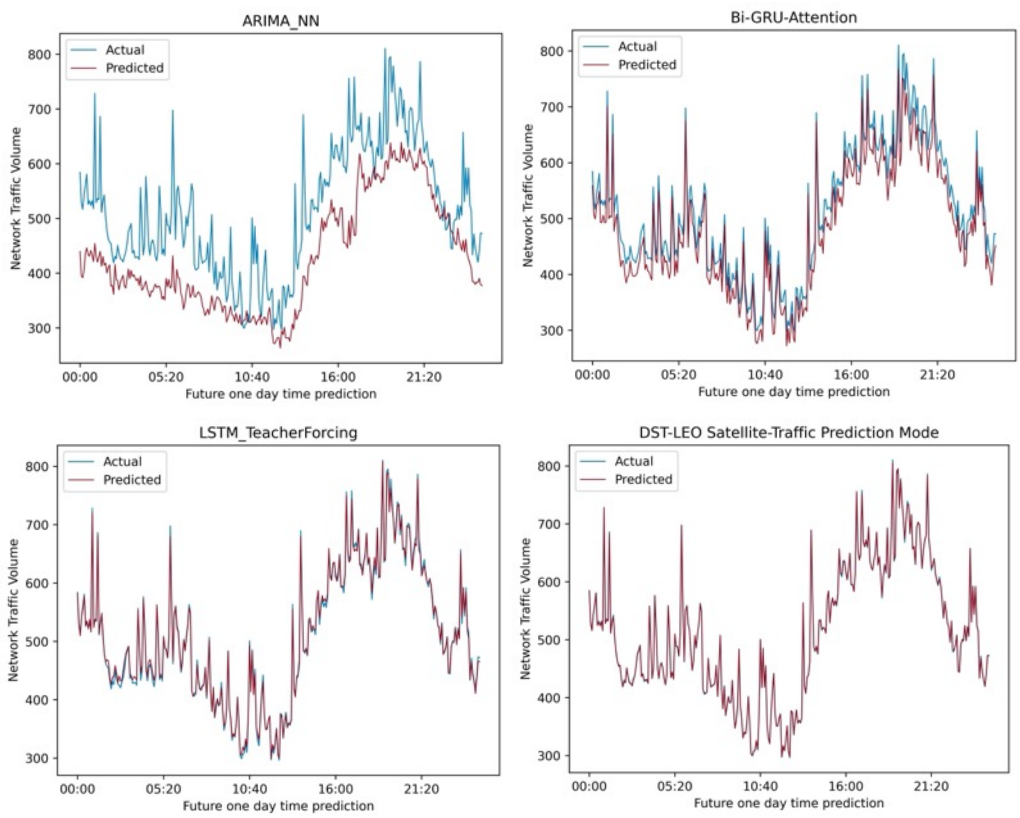

Figure 7. Short-term prediction curve diagram of Link 1 in the Abilene dataset.

In the table, the DST-LEO satellite-traffic prediction model is simplified as DLS-Traffic Prediction.

The authors predicted the traffic change trend for the next week through experiments on the Abilene dataset and compared the performance of the four models. The evaluation metrics are shown in Table 2, and the prediction curve for Link 7 is shown in Figure 8. Link 7 was selected here because the traffic volume changed significantly during the prediction period. The recurrent neural network models performed well on other links. By observing this link, we could see how the models performed when the prediction target changed significantly, which is consistent with the characteristic of large-scale changes in the service volume of low Earth orbit constellations within the same time period. As the number of prediction time steps increased, ARIMA not only failed to show good prediction results in terms of amplitude, but also failed to even predict the trend as well as in the short-term prediction. Therefore, this model is not suitable for this task. The BiGRU with an attention mechanism also did not perform well as the number of prediction time steps increased. It failed to predict the significant change in traffic amplitude on the fifth day, and the traffic prediction results significantly deviated from the true values in other time periods. Although the LSTM with the teacher-forcing mechanism did not have the support of an attention mechanism, since this mechanism directly uses elements from the true target sequence as the input for the next time step with a certain probability during the training process, it can timely correct the large deviations caused by the prediction results of the previous moment during training. Thus, it showed superior performance compared with the previous two models. The DST-LEO satellite-traffic prediction model can extract features of the terrestrial network from different levels due to its spatio-temporal attention mechanism. However, unlike the dataset constructed by the internal Low Earth Orbit constellation business simulation system, this dataset does not contain two types of traffic. As a result, the data separation and fusion techniques may not have fully realized their potential, leading to minor discrepancies between the predicted traffic volume and trends and the actual values at certain moments.

Table 2. Evaluation metrics for the long-term prediction of the Abilene dataset.

Dataset	Abilene			
Models	**MSE**	**RMSE**	**MAE**	**MAPE**
ARIMA [25]	0.0292	0.1709	0.1404	154.6146
BiGRU-Attention [26]	0.0002	0.0141	0.0072	10.0634
LSTM-Teacher Forcing [27]	5.95×10^{-5}	0.0077	0.0052	6.2994
DLS-Traffic Prediction	2.98×10^{-5}	0.0054	0.0039	4.2922

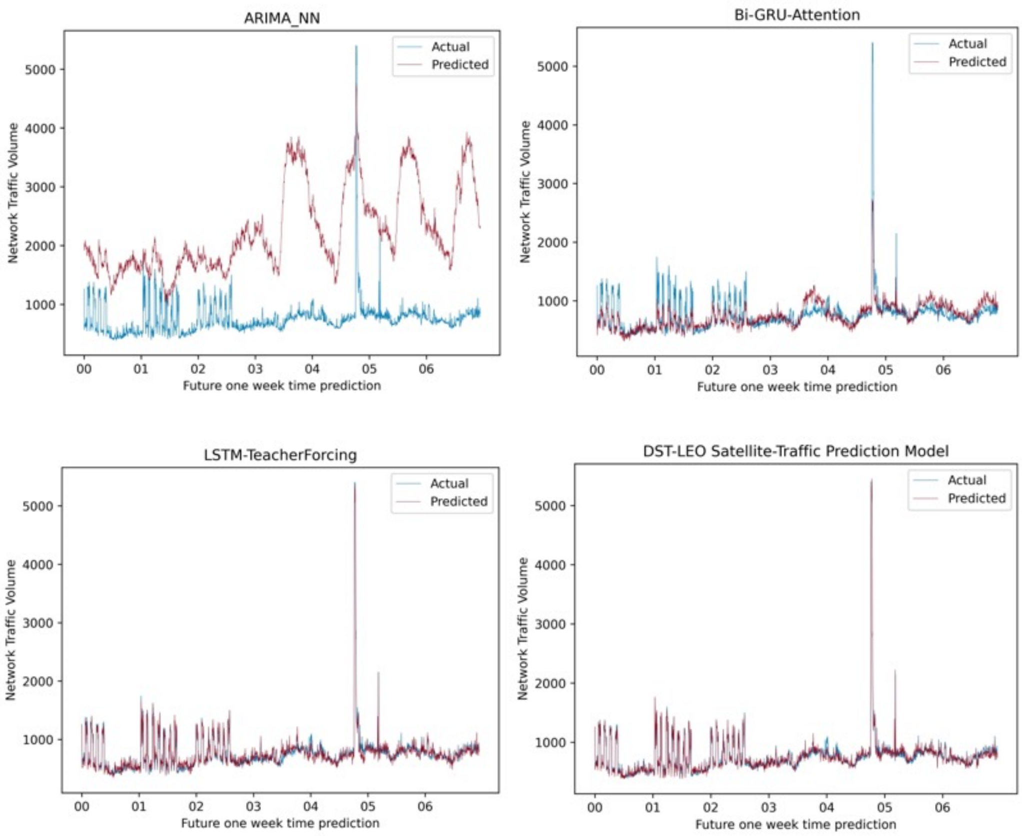

Figure 8. Long-term prediction curve diagram of Link 7 in the Abilene dataset.

Ablation experiments: The authors verified the impact of each module on the target task by removing the complex structure of the model (replacing LSTM with GRU), the bidirectional model design, GCN, the data separation and fusion module, and the spatio-temporal attention mechanism, respectively, and then verified the effectiveness of different modules. The DST-LEO satellite-traffic prediction model was trained on the Abilene dataset after removing the corresponding modules to predict the traffic changes for the next week. The corresponding indicators were as follows. As can be seen from Table 3, different modules played roles in different aspects. It can be seen that GCN and the spatio-temporal attention mechanism had a relatively large impact on the model performance. For networks with topological structures, these two modules can effectively process graph-structured data, deeply mine node feature information, and enhance the model's ability to handle dynamically changing data. The model complexity, bidirectional design, and data separation and fusion enhanced the model's fitting ability. All modules worked together to ultimately achieve an accurate traffic prediction task.

Table 3. Evaluation metrics for the long-term prediction of each model on the Abilene dataset after removing different modules in the ablation experiment.

Dataset	**Abilene**			
Modules	**MSE**	**RMSE**	**MAE**	**MAPE**
GRU	0.0003	0.0168	0.0117	9.1156
BiLSTM	0.0007	0.0257	0.0176	12.5431
GCN	0.0125	0.1120	0.0876	110.5250
Data separation	0.0009	0.0298	0.0220	17.1589
Attention	0.0030	0.0548	0.0413	35.5827

After the authors verified the superior performance of the model through the Abilene dataset, the next step was to use the simulated Low Earth Orbit (LEO) constellation network traffic data collected from the internal LEO constellation service simulation system for testing, predicting the traffic at different time intervals. The internal LEO constellation service simulation system closely replicates the real-world situation. Unlike terrestrial networks, it specifies the communication rules between adjacent satellites. Two satellites can communicate via direct inter-satellite links or through relay satellites, and its traffic has complex spatio-temporal characteristics. On the one hand, the traffic shows periodic, trend-based, and random changes over time. For example, during specific time periods, frequent user activities in certain regions lead to periodic peaks in the satellite traffic of the corresponding areas. Meanwhile, with the development of services and the growth in the user population, the traffic may exhibit an upward or downward trend. On the other hand, there is a spatial correlation among the traffic of different satellites. The traffic of adjacent satellites may influence each other because the geographical areas they cover overlap or they are on the same service chain. The spatio-temporal attention mechanism introduced by the authors for the model could not only capture the time-varying patterns of satellite nodes, but also the on–off positional relationships of the dynamic topology. Secondly, the bidirectional design of BiLSTM enabled the model to obtain more comprehensive context information and better handle long-distance dependencies. The data separation and fusion module designed by the authors could effectively capture when a satellite will disconnect the inter-satellite link with surrounding satellites by separately learning the traffic and link relationships, which helps the traffic prediction capture the trends of sharp increases and decreases. The evaluation metrics are shown in Table 4, and the prediction curves are shown in Figure 9. Both the trends and amplitudes were well-predicted. When combined with key technologies such as traffic classification, load balancing, and traffic scheduling, health monitoring of the LEO constellation can be achieved, ensuring the quality of service for users.

Table 4. Evaluation metrics for the long-term and short-term prediction on the dataset of the internal LEO constellation business simulation system.

Dataset	**Internal Low Earth Orbit Constellation Service Simulation System**			
DLS-Traffic Prediction	**MSE**	**RMSE**	**MAE**	**MAPE**
One day	1.18×10^{-5}	0.0034	0.0026	---
One week	8.48×10^{-6}	0.0029	0.0022	---

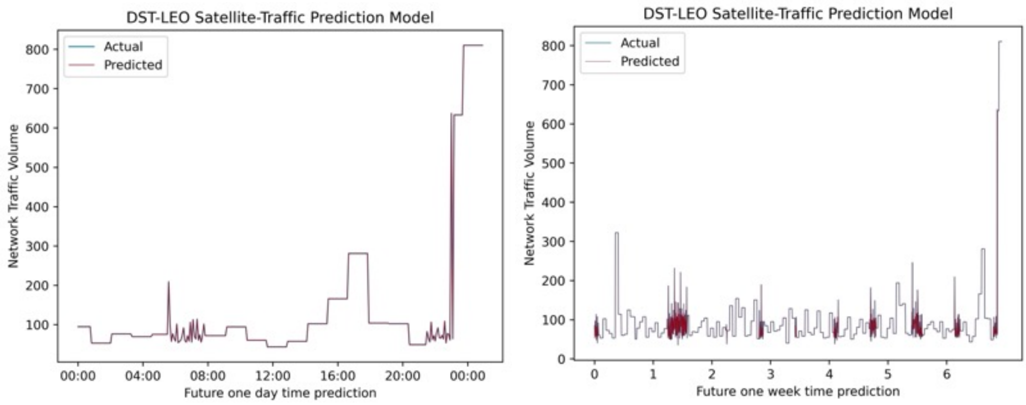

Figure 9. The prediction curves of the DLS traffic prediction model for the long-term and short-term traffic in the internal LEO constellation service simulation system.

5. Discussion

In this section, the authors will focus on discussing some limitations of the DST-LEO satellite-traffic prediction model as well as methods for improving the model to make its predictions more accurate or the model more lightweight. First, the DST-LEO satellite-traffic prediction model achieved excellent prediction results. The team led by the authors is ready to deploy the model at the gateway station in the low Earth orbit satellite Internet system for satellite-to-ground joint testing. The gateway station is equipped with various powerful servers, and the computing power conditions support the deployment of the model. However, the most ideal scenario is still to directly deploy the model on the satellite. Since the model is not extremely large, even if the resources on the satellite are limited, deployment can still be achieved by means of distributed computing and other methods. This is the ideal direction for future attempts at model deployment.

Data-related limitations: The dataset of the internal low Earth orbit constellation business simulation system is too clean, as there may be a large number of noise, missing values, and incorrect records in the real data. Additionally, the existing data collection systems may not comprehensively cover all time and space ranges of the low Earth orbit (LEO) constellation. For example, the traffic data in some remote areas may be insufficiently collected.

Model-related limitations: Although the model successfully completed the designed experiments, there is still room for improvement in its structure. For instance, the teacher-forcing mechanism has provided the authors with a good idea. Additionally, according to the long-term and short-term predictions, a model with a simpler structure can be selected for short-term prediction training, which is easier to deploy and consumes fewer resources.

Future research directions: Techniques such as parallel computing and distributed computing can be adopted to improve the model's processing efficiency. The ultimate goal of the authors' research on traffic-related issues is to design a large-scale model suitable for the satellite communication field. By invoking various top-level small models, different downstream tasks in various scenarios can be completed to achieve system intelligence. For a low Earth orbit constellation system, computing power and memory resources pose significant challenges. In the follow-up, the authors will focus on researching more lightweight models while ensuring the model's performance, aiming to achieve relevant communication and monitoring functions at the lowest cost.

6. Conclusions

This paper proposed a network traffic prediction model, the DST-LEO satellite-traffic prediction model, which is suitable for low Earth orbit (LEO) constellations. By deploying this model in the low Earth orbit (LEO) constellation, accurate network traffic prediction for the LEO constellation can be achieved. This can optimize resource allocation, enhance the performance of the LEO constellation network, reduce unnecessary costs in operation management, and enable the system to adapt to the development of future businesses. The model mainly relies on the graph convolutional network (GCN) and long short-term memory (LSTM) to learn the relationship between the graph topology of the LEO constellation and the traffic time-series. The data separation and fusion module introduced by the authors can split the collected traffic and link status data and send them to the GCN and LSTM layers for training, respectively. Finally, the data are fused to a specific dimension to deeply explore the information contained in different data and discover the impact of link on–off caused by satellite position changes on traffic volume, which can more effectively improve the model's performance. The spatio-temporal attention mechanism introduced by the authors is different from the ordinary attention mechanism. The spatio-temporal attention mechanism enables the model to learn the importance of traffic information at each moment and process the LEO satellite traffic data involving space–time. By capturing the spatio-temporal dependencies in the data, it extracts important features, and all modules work together to complete a very accurate prediction task. In the comparison with other models, it was found that introducing an adaptive mechanism or a teacher-forcing mechanism may potentially improve the performance of the model. This represents a possible direction for future model development. However, the authors have not studied low-cost deployment. If the model is to be deployed on satellites, it is necessary to research more lightweight models or perform distributed computing on different satellites. This is the focus of future research, and is an important challenge for ensuring the quality of user services.

Author Contributions: Conceptualization, K.L., Y.Z. and S.L.; Methodology, K.L.; Software, K.L.; Validation, K.L.; Formal analysis, K.L., Y.Z. and S.L.; Investigation, K.L. and S.L.; Data curation, K.L.; Writing—original draft, K.L.; Writing—review and editing, K.L., Y.Z. and S.L.; Visualization, K.L.; Supervision, Y.Z. and S.L.; Project administration, K.L. and S.L.; Funding acquisition, S.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Key Research and Development Program of China (2022YFB2902505).

Data Availability Statement: The dataset provided in this article is not easily accessible, as the data is part of an ongoing research project. Requests for access to the dataset should be sent directly to 18131790215@163.com.

Acknowledgments: Kexuan Liu would like to thank Yasheng Zhang and Shan Lu for their support and guidance during their research over the past two years and also appreciates the valuable feedback from the editors and reviewers on this article. The authors wish everyone all the best.

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

References

1. Zhang, Y.; Wang, Y.; Wu, J. Review of the Development of Foreign Communication Satellites in 2023. *Space Int.* **2024**, *4*, 21–27.
2. Liu, Y.; Su, Y.; Zhou, Y.; Cao, H.; Shi, J. Frequency Offset Estimation for High Dynamic LEO Satellite Communication Systems. In Proceedings of the 2019 11th International Conference on Wireless Communications and Signal Processing (WCSP), Xi'an, China, 23–25 October 2019; pp. 1–6. [CrossRef]
3. Chan, S.; Lee, H.; Kim, S.; Oh, D. Intelligent Low Complexity Resource Allocation Method for Integrated Satellite-Terrestrial Systems. *IEEE Wirel. Commun. Lett.* **2022**, *11*, 1087–1091. [CrossRef]
4. Lysenko, O.; Yavisya, V.; Sushyn, I. Method for Ensuring the Quality of Service of Geostationary Satellite Communication Systems. In Proceedings of the 2021 IEEE 8th International Conference on Problems of Infocommunications, Science and Technology (PIC S&T), Kharkiv, Ukraine, 5–7 October 2021; pp. 353–356. [CrossRef]
5. Zhou, Z. Research on Traffic Flow Prediction Model Based on Artificial Neural Network. Master's Thesis, Harbin Engineering University, Harbin, China, 2007.

6. Shi, S. Research on Network Traffic Prediction and Application Based on Machine Learning. Master's Thesis, Beijing University of Posts and Telecommunications, Beijing, China, 2019.
7. Duan, W.; Gong, Z. Research and application of urban real-time traffic flow prediction based on STARIMA. In Proceedings of the 2022 IEEE International Conference on Advances in Electrical Engineering and Computer Applications (AEECA), Dalian, China, 20–21 August 2022; pp. 1521–1525. [CrossRef]
8. Madan, R.; Mangipudi, P.S. Predicting Computer Network Traffic: A Time Series Forecasting Approach Using DWT, ARIMA and RNN. In Proceedings of the 2018 Eleventh International Conference on Contemporary Computing (IC3), Noida, India, 2–4 August 2018; pp. 1–5. [CrossRef]
9. Ye, Q.; Yang, X.; Chen, C.; Wang, J. River Water Quality Parameters Prediction Method Based on LSTM-RNN Model. In Proceedings of the 2019 Chinese Control And Decision Conference (CCDC), Nanchang, China, 3–5 June 2019; pp. 3024–3028. [CrossRef]
10. Han, P.; Wang, W.; Shi, Q.; Yang, J. Real-time Short- Term Trajectory Prediction Based on GRU Neural Network. In Proceedings of the 2019 IEEE/AIAA 38th Digital Avionics Systems Conference (DASC), San Diego, CA, USA, 8–12 September 2019; pp. 1–8. [CrossRef]
11. Zhao, L.; Song, Y.; Zhang, C.; Liu, Y.; Wang, P.; Lin, T.; Deng, M.; Li, H. T-GCN: A Temporal Graph Convolutional Network for Traffic Prediction. *IEEE Trans. Intell. Transp. Syst.* **2020**, *21*, 3848–3858. [CrossRef]
12. Zhu, J.; Lin, Y.; Hu, C. Integrated Technology of Laser Communication and Ranging for High—Dynamic Inter—Satellite Links. *J. China Acad. Electron. Inf. Technol.* **2024**, *19*, 598–605+614.
13. Wang, L.; Tong, K.; Yang, J.; Chen, X. An on-orbit calibration method for time delay of inter-satellite link transponder in tracking and data relay satellite. *Electron. Lett.* **2023**, *59*, e12906. [CrossRef]
14. Lahat, D.; Adali, T.; Jutten, C. Multimodal Data Fusion: An Overview of Methods, Challenges, and Prospects. *Proc. IEEE* **2015**, *103*, 1449–1477. [CrossRef]
15. Li, H.; Mu, D.; Wang, P.; Li, Y.; Wang, D. Prediction of Obstetric Patient Flow and Horizontal Allocation of Medical Resources Based on Time Series Analysis. *Front. Public Health* **2021**, *9*, 646157. [CrossRef] [PubMed]
16. Yang, X.; Zhai, P.; Shi, L.; Yu, X.; Qu, X.; Chen, Y. Prediction of mine water flow based on singular spectrum analysis and multiple time-series coupled model. *Arab. J. Geosci.* **2021**, *14*, 2858. [CrossRef]
17. Saha, S.; Haque, A.; Sidebottom, G. Multi-Step Internet Traffic Forecasting Models with Variable Forecast Horizons for Proactive Network Management. *Sensors* **2024**, *24*, 1871. [CrossRef] [PubMed]
18. Rau, F.; Soto, I.; Zabala-Blanco, D.; Azurdia-Meza, C.; Ijaz, M.; Ekpo, S.; Gutierrez, S. A Novel Traffic Prediction Method Using Machine Learning for Energy Efficiency in Service Provider Networks. *Sensors* **2023**, *23*, 4997. [CrossRef] [PubMed]
19. Sergi, M.; Esther, S.; Enric, P. RNN-CNN Hybrid Model to Predict C-ATC CAPACITY Regulations for En-Route Traffic. *Aerospace* **2022**, *9*, 93. [CrossRef]
20. Yu, Z.; Chang, C.; Li, T. LSTM-based Traffic Prediction Method for Named Data Networking. *Chang. Inf. Commun.* **2022**, *35*, 93–95.
21. Jiang, W. Internet traffic matrix prediction with convolutional LSTM neural network. *Internet Technol. Lett.* **2021**, *5*, e322. [CrossRef]
22. Li, N.; Hu, L.; Deng, Z.L.; Su, T.; Liu, J.W. Research on GRU Neural Network Satellite Traffic Prediction Based on Transfer Learning. *Wirel. Pers. Commun.* **2021**, *118*, 815–827. [CrossRef]
23. Lü, G.; Wang, K. Dynamic Traffic Prediction in SDN Networks Based on Spatio-Temporal Graph Attention Mechanism. *Comput. Eng. Appl.* **2025**, 1–8.
24. Liu, Z.; Li, W.; Feng, J.; Zhang, J. Research on Satellite Network Traffic Prediction Based on Improved GRU Neural Network. *Sensors* **2022**, *22*, 8678. [CrossRef] [PubMed]
25. Musarat, M.A.; Alaloul, W.S.; Rabbani, M.B.A.; Ali, M.; Altaf, M.; Fediuk, R.; Vatin, N.; Klyuev, S.; Bukhari, H.; Sadiq, A.; et al. Kabul River Flow Prediction Using Automated ARIMA Forecasting: A Machine Learning Approach. *Sustainability* **2021**, *13*, 10720. [CrossRef]
26. Zhou, T.; Huang, B.; Li, R.; Liu, X.; Huang, Z. An attention-based deep learning model for citywide traffic flow forecasting. *Int. J. Digit. Earth* **2022**, *15*, 323–344. [CrossRef]
27. Lu, C. Performance analysis of attention mechanism and teacher forcing ratio based on machine translation. *J. Phys. Conf. Ser.* **2023**, *2580*, 012006. [CrossRef]