



TC–GAT: A Temporal Convolutional Graph Attention Network for Traffic Flow Prediction

by

Yang Lu

Supervisor: Dr Gordon Dickers

Project submitted as part of the requirements for the
award of MSc Data and Artificial Intelligence

September 2023

DECLARATION

I, ...Yang Lu.....
declare that I am the sole author of this Project; that all references cited have been consulted; that I have conducted all work of which this is a record, and that the finished work lies within the prescribed word limits.

This work has not previously been accepted as part of any other degree submission.

Signed : 

Date : September 7, 2023.....

FORM OF CONSENT

I Yang Lu..... hereby
consent that my Project, submitted in candidature for the M.Sc. Data and Artificial Intelligence
degree, if successful, may be made available for inter-library loan or photocopying (subject to the
law of copyright), and that the title and abstract may be made available to outside organizations.

Signed : 

Date : September 7, 2023.....

Abstract

With the development of society, the number of cars continues to increase, and traffic congestion has gradually become the main problem hindering urban development. Traffic flow data records the number of traffic entities passing through a certain section of the road for a period of time, and its real-time and accurate prediction plays an important role in improving traffic congestion. Therefore, it is of great significance to study the theory and method of traffic flow forecasting to better utilize the performance of urban traffic road network. However, due to the complex pattern of traffic flow changing rapidly in time and space, it is difficult for existing prediction methods to accurately capture the spatio-temporal coupling characteristics of traffic data. Therefore, based on the research and analysis of urban road traffic data, this thesis proposes a traffic flow prediction model TC-GAT (Temporal Convolutional Graph Attention Network) based on Temporal Convolutional Network (TCN), Graph Attention Network (GAT), and Transformer. In order to solve the problem that Graph Convolution Network (GCN) can only statically fuse the features of road network nodes, TC-GAT uses GAT to dynamically aggregate spatial features to model the spatial dependence of traffic flow. Aiming at the poor long-term prediction effect of traffic flow, TCN is used to model temporal dependence by capturing the dynamic change of traffic flow on roads. In addition, the highly parallelized self-attention mechanism in Transformer is introduced to improve prediction accuracy by adjusting the importance of different time points and assembling global temporal information. The experimental results show that TC-GAT achieves better prediction performance than the comparison model. Meanwhile, the necessity of each module in the model is evaluated by ablation experiments.

Content

Abstract	3
Content	4
CHAPTER 1: INTRODUCTION	5
CHAPTER 2: LITERATURE REVIEW	7
2.1 Temporal feature dependence extraction	7
2.2 Spatial feature dependence extraction	9
2.3 Graph Structure Learning	10
2.4 Attention Mechanism	11
CHAPTER 3: Methodology	12
3.1 Traffic flow knowledge	12
3.1.1 Temporal dependence.	12
3.1.2 Spatial dependence.	13
3.1.3 Problem Definition.	14
3.2 Framework	15
3.2.1 Spatial-Dependent Feature Extraction Module	15
3.2.2 Temporal-Dependent Feature Extraction Module	17
3.2.3 Attention mechanism layer	18
CHAPTER 4: Experimental Design	20
4.1 Data Description	20
4.2 Data Processing	21
4.2 Evaluation Metrics	21
4.3 Loss Function	22
4.4 Baselines	23
4.6 Experimental settings	24
CHAPTER 5: Results & Discussions	25
CHAPTER 6: Conclusions and Recommendations	26
CHAPTER 7: REFLECTION	28
REFERENCES	30

CHAPTER 1: INTRODUCTION

In sync with the rapid development of Transportation Systems (ITS) [1], traffic flow prediction is gaining popularity because of its potential to provide valuable data support for urban planning, including road construction, public transportation planning, to help cities achieve a rational transportation layout and development. Accurate traffic flow prediction can help traffic management departments and drivers to understand the future road conditions, so as to reasonably plan travel routes and time, reduce traffic congestion, improve road use and transportation efficiency, and thus reduce energy consumption and environmental pollution. In addition, traffic flow prediction provides important data support for urban transportation planning, which can help planners optimize road layout, public transportation routes and transportation facilities, and improve the overall efficiency and sustainability of urban transportation systems. At the same time, accurate traffic flow prediction can help travelers plan the best travel plan, choose the shortest route and the most suitable travel time, reduce travel time and uncertainty, and improve the comfort and convenience of travel. Hence, traffic flow prediction can help traffic management and drivers to understand traffic conditions in advance and avoid traffic accident-prone areas and high-risk roads, thus reducing traffic accidents and improving traffic safety.

Traffic data can be regarded as a kind of time-series data, but compared with other time-series prediction problems, traffic flow prediction is more challenging, which is mainly manifested by the complex spatio-temporal correlation of traffic data and the influence of some external factors [2]. The spatio-temporal correlation of traffic data includes complex spatial dependence and dynamic time dependence. Spatially, different neighboring nodes in the traffic network have different impacts on other nodes, and the impacts of the same node on other predicted locations change with time; in terms of time, the observed data of the same node at different time steps are nonlinear, and the influence between multiple time steps is often long-term. At the same time, traffic data usually show periodicity, such as seasons, weekdays and days off. External influences on traffic data mainly include weather states, emergencies, and road attributes. In summary, traffic data exhibit complex dynamic correlations in both time and space, and how to tap into such nonlinear spatio-temporal connections for accurate traffic prediction is an important topic.

Significant advancements have been achieved in the field of traffic forecasting after years of dedicated work and extensive study. The existing approaches for predicting traffic flow may be categorized into two main groups: conventional techniques and deep learning methods. The conventional approaches mostly include traditional statistical techniques and machine learning methodologies. Classical statistical approaches use data-driven statistical models, such as the Historical Average (HA) [3] and the Auto-Regressive Integrated Moving Average (ARIMA) [4], to make predictions about the data. Nevertheless, these methodologies are often limited in their applicability to datasets of a relatively small scale and struggle to effectively capture intricate nonlinear interactions. Consequently, they fail to fulfill the prevailing requirements for traffic prediction. In order to tackle the aforementioned issues, several machine learning techniques have been suggested by academics, such as Support Vector Regression (SVR) [5] and Random Forest Regression (RFR) [6]. Nevertheless, machine learning techniques heavily rely on feature engineering and often need the creation of supplementary manual features beforehand by domain experts. This aspect poses a challenge when attempting to extract intricate spatio-temporal patterns. Researchers are increasingly utilizing deep learning techniques to investigate traffic prediction in light of the expanding datasets in the transportation domain and advancements in artificial intelligence technology.

These methods include recurrent neural networks such as Long Short-Term Memory (LSTM) and Gate Recurrent Unit (GRU), as well as the Seq2seq model, which is based on encoding and decoding structures [7, 8]. Nevertheless, conventional shallow deep learning networks encounter challenges in accurately predicting outcomes in complicated traffic scenarios. Additionally, previous prediction techniques have failed to include spatial dependency models, leaving opportunity for enhancing the accuracy of predictions.

The use of graph neural networks has been more prevalent, and one often employed method for capturing spatial correlation is the Graph Convolution Network (GCN) [9]. GCN employs a predetermined and unchanging adjacency matrix to represent the connections between traffic nodes. However, this approach fails to accurately assess the significance of various traffic nodes. Moreover, apart from the spatial correlation between neighboring nodes, there may exist a certain level of correlation between distant nodes within the urban area. This correlation implies that these nodes exhibit similar temporal patterns. Furthermore, apart from the spatial correlation seen among adjacent nodes, it is possible that some nodes located at a distance from each other inside the city exhibit correlation, indicating a similarity in their temporal patterns. Furthermore, the efficacy of current approaches for acquiring knowledge on temporal dynamics is limited. One of the challenges encountered by Recurrent Neural Networks (RNNs) is the issue of gradient vanishing, which hinders their ability to effectively capture the temporal dependencies present in traffic data over extended time periods.

Hence, based on the above description, we propose a novel traffic flow prediction method called the Temporal Convolutional Graph Attention Network (TC-GAT), which uses Graph Attention Network (GAT) [10] to capture spatial dependence and Temporal Convolutional Network (TCN) [11] to model temporal dependence also we adopt the Attention mechanism^[12] to adjust the importance of different traffic node. To the greatest of our knowledge, this is the first attempt to integrate the TCN and the GAT. We believe that our pioneering effort will cast light on how to enhance the accuracy of traffic flow prediction. I evaluate this model by adopting the taxi speed data of the Luohu District in Shenzhen and Los-loop datasets. The experimental results show that under different evaluation metrics, the performance of TC-GAT has a good improvement compared to other models.

The subsequent sections of this work are structured in the following manner. In the second chapter, a complete literature review is conducted on four topics: extraction of temporal feature dependency, extraction of spatial feature dependence, learning of graph structure, and attention mechanism. CHAPTER 3 of the dissertation presents a comprehensive analysis of the issue by formulating it mathematically. Additionally, it provides a detailed explanation of the fundamental technologies used within the model. Furthermore, the chapter delves into an extensive discussion of the overall architecture of the model. CHAPTER 4 provides a comprehensive account of the data collecting and experimental background. Subsequently, the experimental design is outlined, including baselines and evaluation processes. Subsequently, an analysis was conducted on the experimental data presented in CHAPTER 5, whereby a comparison was made between the obtained findings and established baselines in order to ascertain the soundness of the proposed model. In the sixth chapter, a comprehensive evaluation of the project was conducted, including an examination of possible improvements to the existing model and an exploration of future research endeavors. Finally, I discussed my comprehensive work experience in CHAPTER 7 and provided a summary of my diligent efforts in conducting the experiment.

CHAPTER 2: LITERATURE REVIEW

The prediction of traffic flow relies on the analysis of spatio-temporal data obtained from diverse sensors. This prediction task encompasses both spatial and temporal characteristics, thus qualifying as a spatio-temporal prediction task. Recent research efforts have primarily concentrated on enhancing prediction accuracy by addressing these two aspects. In the context of temporal feature extraction, the field is primarily categorized into three main approaches: traditional statistical theory, machine learning, and deep learning methods. On the other hand, when considering spatial feature extraction in the domain of traffic networks, recent modeling methods can be classified into two categories: those based on Euclidean space and those based on non-Euclidean space. Since non-Euclidean space-based methods mainly use graph convolution to model spatial dependencies, but the traditional predefined adjacency matrix cannot adequately express the spatial connections of traffic nodes, so graph structure learning has also become a hot spot of researchers' attention.

2.1 Temporal feature dependence extraction

Research on time-dependent modeling of traffic flow has a history of several decades. Early research on traffic flow was mostly based on traditional statistical methods and focused on temporal modeling. After the development of computer hardware, various types of machine learning methods and deep learning methods have been applied to traffic flow prediction, and in view of the strong expressive ability of deep learning, deep learning-based methods have become the focus of researchers.

Among the methods based on traditional statistical theory, the Kalman Filter (KF) [14] based method uses a series of time-varying observations containing noise and other errors to produce estimates of unknown variables; such methods are difficult to deal with the stochastic and cyclical nature of traffic data, and their application in the field of traffic flow prediction has major limitations. HA [3] uses a set of recent historical data to predict the future period or periods of data. This method is simple and easy to implement, and has good prediction effect in the case of time series without seasonality and periodicity, but it is weak on time series data with obvious periodicity. The ARIMA-based method [4] adds the operation of difference on the basis of ARMA, and its basic idea is autoregression, i.e., the value of the current point is regressed from the value of the previous several adjacent points; ARIMA model fits each sequence individually, which is less efficient in dealing with the large-scale time series, and due to the fact that each sequence is fit individually and independently, it is not possible to share the similarity law that exists among the different scenes.

Among the machine learning-based methods, SVR model-based methods [5] predict future traffic data by adjusting the regression plane so that the sum of the distances from the training data to the plane is the closest; however, SVR is not suitable for large datasets, and it performs poorly when there are many error data in the traffic dataset, or when the dimension of the features of a single data point is larger than the sample size of the data. The prediction method based on Bayesian network [15] analyzes the neighboring information between adjacent roads through mathematical statistics to extract the spatial characteristics of the traffic data and determine the future trend of the current prediction point; however, due to the expressive ability of machine learning, Bayesian network is difficult to achieve good modeling effect for large-scale datasets. The K-nearest Neighbors Algorithm (KNN) based method [16] uses Euclidean distance, Manhattan distance and other distance metrics to predict the future trend of the traffic

data. KNN does not require data input assumptions, and the outliers in the data have less influence on the KNN model, but its computational and spatial complexity is high.

With the development of artificial intelligence technology and the improvement of related hardware, deep learning is increasingly used for traffic flow prediction. Dougher firstly applied neural networks to the task of traffic flow prediction, and Polson et al. established a deep learning architecture for traffic flow prediction [9]. After the emergence of RNNs for capturing time series dependence and Convolutional Neural Network (CNNs) for extracting spatial features, related studies have gradually combined CNNs and RNNs for traffic flow prediction, but the models that simply combine the two have yet to be improved in terms of obtaining the spatial and temporal dependence of high-dimensional traffic data.

In terms of capturing temporal dependencies, RNNs can preserve the previously input sequence information with their internal vectors, which is widely used in the field of traffic flow prediction [17]. However, RNNs still have some drawbacks. First, RNNs-based methods cannot process time series in parallel, which increases the training time. Second, when RNNs are combined with graph convolutional networks, it is more likely to encounter the problem of gradient vanishing [18].

In response to the above limitations of RNNs, deep learning methods such as CNNs and attention mechanisms have been used in several studies to model temporal dependencies. CNNs have been favored by researchers in recent years because of their ability to process time-series data in parallel. Among them, Extended Convolutional Networks are a special form of CNNs, consisting of null CNNs, which do not have to be processed sequentially like RNNs, and gradient backpropagation does not keep accumulating along the time dimension, thus avoiding the problem of gradient dispersion. Due to the above advantages, more and more studies have used extended convolution for traffic flow prediction, and STFGNN [19] and GraphWavenet [20] use parallel gated expansion CNNs to capture the temporal properties of traffic data from different dimensions through a gating mechanism and an expansion factor that increases with the number of layers. However, the capture of long-term dependencies still requires stacking a large number of expanded convolutional layers to increase the sensory field of the CNN, and how to better capture long-term dependencies is still an area that requires in-depth research. Recently, The proposal of the TCN [11] proves that a simple convolutional network is more effective than the RNN across a variety of tasks of late. Hence, in [21], a novel local temporal convolutional mechanism is proposed to capture the long-term local dynamics effectively.

Attention mechanisms, with their ability to learn temporal inter-signal dependencies, have been widely introduced into automatic coding, deep convolutional and generative networks with favorable results. In time series prediction studies, the attention mechanism has been shown to be an effective technique for extracting data features. For example, Wu et al [22] proposed a multi-component attention approach for time series prediction, in which a bidirectional LSTM model is used to capture seasonal trends and temporal features, a CNN model is used to capture local trend features of residual units, and the introduction of an attention module allows the multi-component streaming data to connect highly correlated historical information in the final prediction. ASTGCN [23] and DGCN [24] use the attention mechanism to directly correlate links between traffic data at different time points, thus improving the model's ability to perceive long-term series. Google proposed to utilize the self-attention mechanism to replace the traditional recurrent neural network in Transformer, which has been successfully applied in the field of natural language processing. Compared with other time-series prediction methods, the auto-attention mechanism is better at capturing the internal correlation of time-series data, and the intrinsic structure of CNNs is hardly better than autoregressive models like the auto-attention mechanism [25], and the auto-attention mechanism shows outstanding advantages in long-term dependency capturing.

2.2 Spatial feature dependence extraction

In terms of capturing spatial dependence, some studies [26-27] modeled the traffic network as a regular two-dimensional grid based on Euclidean space, and the two-dimensional traffic grid data can be processed by CNN to capture spatial dependence because CNN has stable and invariant properties, and can extract the spatial features of the image very well. However, the topology does not have the translation invariance of the image, so the traditional convolution method is no longer applicable, and the research on traffic flow prediction based on non-Euclidean space modeling has gradually become a hot spot.

In the field of traffic flow prediction using Euclidean space, several models have been proposed. The Conv-LSTM model, as described in [28], begins by mapping the traffic flow information into a one-dimensional vector space. It then combines these one-dimensional vectors to form a traffic information matrix. By utilizing a combination of Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM), the model is able to extract both spatial and temporal features from the traffic data. Another model, known as ST-ResNet [29], takes a different approach. It divides the trajectory information of cities into grid maps, which represent different regions based on their geographic locations. The model then counts the inflow and outflow of each entrance and transforms this information into a two-channel matrix. This matrix is processed in a manner similar to image processing techniques. Zhang et al. [30] propose a model that represents the grid maps as weighted directed graphs. This is achieved by dividing the actual geographic information into grid maps and then performing a graph embedding operation. The resulting graphs are then inputted into a CNN for spatial feature learning. The RSTN model [31] integrates CNN, LSTM, and Conv-LSTM modules using residual networks to effectively capture spatio-temporal relationships. It approaches the conventional traffic prediction issue by considering it as a residual function of trip density within certain time intervals. The study conducted by MGSTC [27] investigates various degrees of spatio-temporal correlations by using numerous spatio-temporal convolutional branches. Additionally, the study dynamically integrates spatio-temporal variables with external parameters in order to enhance the performance of the model's predictions.

Graph convolution is often used in traffic flow prediction research that utilizes non-Euclidean spatial modeling to effectively capture spatial correlations. There are two primary categories of graph convolution, namely the spectral-based methodology [32] and the spatial-based methodology [33]. The STGCN approach [34] employs a generalized graph structure to effectively represent the traffic network. This involves modeling network stations as nodes within the graph. Each node in the graph contains a collection of observations recorded at various time intervals. Spatial information is extracted using Graph Convolutional Networks (GCN), while temporal features are captured using Gated Recurrent Units (GRU). In a similar vein, Chai et al. (35) utilize a multi-graph convolutional network to forecast traffic flow. They construct multiple graphs based on factors such as distances, traffic interactions, and correlations between stops within the traffic network. These graphs are then fused together, enabling the learning of spatial-temporal features through the combined use of GCN and Encoder-Decoder networks. However, despite the remarkable predictive accuracy attained by GCN models, a common limitation is their tendency to capture static spatial correlations in traffic data, without accounting for the potential evolution of these dependencies over time. In recent times, GAT has gained popularity in the field of traffic prediction due to its ability to integrate an explicit attention mechanism. This mechanism allows for the dynamic extraction of the topological link between the target street and its nearby streets. As a result, GAT is increasingly being used to represent spatial-dependent features in traffic prediction tasks. The researchers used Geographically Weighted Regression (GWR) to analyze the spatially-dependent characteristics of

roadways. Additionally, a long short-term memory network (LSTM) is used for the purpose of extracting features that are reliant on temporal information.

2.3 Graph Structure Learning

Current graph structure learning mainly obtains the adjacency matrix by calculating the similarity between time series, which can be divided into two categories: static and dynamic graphs. Static graph construction methods aim to use existing knowledge to construct graph structures during preprocessing, and early graph construction methods only use the connectivity between roads to obtain a binary adjacency matrix. To further improve the graph structure, some studies have introduced methods to measure the static similarity of the time series, such as the DTW [37] and the Pearson coefficient method, to obtain static graphs based on the temporal similarity between nodes. degree to obtain the static graph.

The application of dynamic graphs is to further improve the representation ability of the adjacency matrix. Due to the dynamic characteristics of the traffic network, static graphs have natural deficiencies in describing complex road networks, and some studies have introduced an attention mechanism based on the binary adjacency matrix, so as to make the influence weight of the nodes change dynamically, and then to adapt to the dynamic changes of the road network. However, in addition to neighboring nodes, some more distant traffic nodes may also have a certain degree of correlation, so more in-depth research is needed for dynamic graph structure learning. At this stage, the implementation of dynamic graphs mainly relies on two types of methods: node embedding [38] and dynamic time series similarity metric. The performance can be improved by obtaining adaptive adjacency matrix through node embedding, in which the parameters of the adjacency matrix can be learned through the training process; and the dynamic similarity metric captures the dynamics of the traffic data by real-time calculation of the similarity of the traffic sequences and updating the adjacency matrix. The dynamic similarity measure captures the dynamic connection of traffic data by calculating the similarity between traffic sequences and updating the neighbor matrix in real time.

Time series similarity measures are also widely used in graph structure learning. The existing correlation measures can be roughly categorized into the following three types. One is based on Euclidean distance, for sequences of the same length, calculate the distance between every two points, and then sum up, the smaller the distance, the higher the similarity; however, this method can not distinguish between shape similarity, and can not reflect the similarity of the dynamic range of changes in the time series trend. The second is based on pattern distance, this type of method divides the time series into different subsets according to the change trend, and each subset represents a class of change patterns. The third is based on shape distance, such as DTW [37], where DTW first aligns the time series in terms of shape, and then calculates the sum of the distances from points on one sequence to each point on the other sequence by dynamic programming and takes the The similarity of the sequences is measured by calculating the sum of distances from points on one sequence to points on the other sequence by dynamic programming and taking the minimum of the sum of distances, but the generation of neighbor-joining graphs by using the DTW algorithm has high However, the generation of neighbor graphs using the DTW algorithm has a high time complexity.

Although graph neural networks can deal with graph structure data, there are limitations: they can only be applied in static graphs, and it is not feasible to deal with dynamic graphs for the time being; there is a great obstacle to applying graph networks on large-scale data; The network structure is shallow,

generally up to 3 layers are stacked, and too many layers will lead to the problem of oversmoothing. Based on the above problems, researchers later proposed a variety of improved models on the basis of GNN [51], including graph attention network, graph convolutional network, graph isomorphic network, etc. Most of these models have improved the problems of GNN, and the performance of the model has been greatly improved.

2.4 Attention Mechanism

In deep learning, the attention mechanism is equivalent to the human visual attention mechanism [40]. When people obtain information about something, they usually scan the overall information quickly to determine the important parts, and then devote greater attention and energy to the important parts and further analyze them to obtain the corresponding detailed information, which essentially realizes the efficient allocation of information processing resources [41]. This mechanism not only helps people to quickly access the most effective information, effectively saving attention resources, but also according to the actual situation of selective access to some of the information, help people faster and better. This mechanism not only helps people to obtain the most effective information quickly and save attention resources, but also selectively obtain part of the information according to the actual situation, which can help people to complete the in-depth study of important information faster and better, and efficiently process existing information.

The attention mechanism in deep learning plays a crucial role in identifying and prioritizing relevant information pertaining to the current topic from a vast amount of available data [42]. In the field of deep learning, the attention mechanism plays a crucial role. It involves assigning weights to input information, with the intention of emphasizing the significance of certain feature information. By assigning larger weights to more important features, this mechanism prevents the model from excessively prioritizing irrelevant features. Consequently, the attention mechanism effectively reduces the computational and storage costs associated with the model, thereby facilitating the production of more accurate results. Additionally, the attention mechanism contributes to enhancing the model's processing efficiency when dealing with substantial amounts of data. This improvement enables the model to make faster judgments, ultimately leading to efficient utilization of computer resources. In summary, the primary objective of the attention mechanism is to systematically distribute attention resources, effectively eliminate irrelevant information, and provide efficient information selection. This enables the acquisition of information that is more pertinent to the focal point of attention and of greater significance.

Most of the current research on the application of the attention mechanism is mainly based on the Encoder-Decoder model [43]. In the simple Encoder-Decoder model, the Encoder is responsible for inputting the data ($X_1, X_2 \dots, X_n$), encoding the input data into a fixed-length vector and encoding it into a series of hidden states, and then connecting to the last hidden state of the Decoder, and decoding the hidden state into the output ($Y_1, Y_2 \dots, Y_n$), which is a process that requires that there is enough hidden information stored in the hidden states, which is susceptible to the long-distance dependency problem.

Moreover, the Encoder-Decoder structure requires the encoder to compress the input into a single vector of predefined dimensions, regardless of the size or informative content of the input. The act of compressing the model leads to a decrease in its overall performance, leading to the emergence of a phenomena often referred to as the "limited capacity of context vectors." The interpretability of a fundamental encoder-decoder model is considerably constrained as a result of the compression of essential information into a context vector, which lacks a pre-established structure, hence making it ineffectual.

In addition to its integration inside the Encoder-Decoder framework, the attention mechanism's structure may be independently used in other deep learning models to enhance the model's feature representation [44]. The initial application of the attention mechanism was in the context of sequence processing tasks. Its purpose was to determine the level of dependency between the target sequence and the input sequence. By doing so, the attention mechanism enables the concentration of attention on the sequences that are most pertinent to the task at hand. Consequently, it also helps to mitigate the interference caused by sequences that have less relevance to the task. Sequences that exhibit less relevance to the job at hand and mitigate the distraction resulting from such less relevant sequences. The attention mechanism has been used in several fields like data restoration [45], sentiment classification [46], picture segmentation [47], among others, in recent years. Furthermore, the attention mechanism offers exceptional performance and can be employed to extract potential mappings between disparate modalities without the need for explicit supervision. This attention-based model is capable of extracting intricate mappings and dependencies. Hence, this study integrates the attention mechanism into the traffic flow prediction model in order to enhance the quality of the outcomes.

CHAPTER 3: Methodology

This chapter first introduces the spatiotemporal characteristics of traffic flow and mathematically describes the task of traffic flow prediction; Subsequently, the network architecture and implementation principles of TC-GAT will be presented.

3.1 Traffic flow knowledge

3.1.1 Temporal dependence.

Figure 3-1 shows the traffic flow data of Bantian Street, Longgang District, Shenzhen City, five and Jiaxian intersection for ten days, as can be seen in the figure, the same traffic node of the traffic data in many days shows a cyclical nonlinear law of change, but also in the holidays and weekdays with different traffic patterns, and due to the regularity of human social activities, the morning and evening for the travel peak, the traffic flow in the night to reach a low peak. And because of the sensor's measurement error or the impact of emergencies, there is a significant deviation from the cycle curve of the data, in the historical data curve for obvious burrs. In addition, weather, traffic control and other influencing factors further enhance the non-periodicity of traffic flow data.

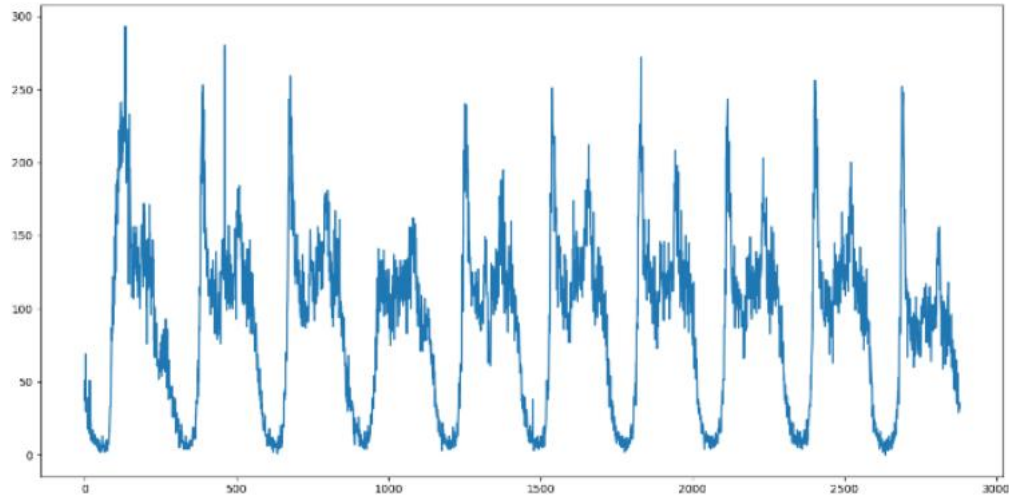


Fig. 3-1 Historical traffic flow at Wuhe-Jiaxian intersection

3.1.2 Spatial dependence.

Traffic flow data is obtained from the measurement of sensors set in different spatial locations in the real world, and the traffic flow sequence measured by a single sensor has temporal autocorrelation, so it can be processed using time series processing methods; the measurement nodes of multiple sensors constitute a complex connectivity system, and the traffic situation of the upstream traffic nodes in the same direction has an obvious influence on the downstream, and the adjacent nodes will have a strong spatial connection, as illustrated in Figure 3-2; however, the more distant nodes may also have similar traffic patterns due to similar social functions, such as highway entrances and exits, shopping malls and hospitals. spatial connection; however, more distant nodes may also have similar traffic patterns due to similar social functions, such as highway entrances and exits, shopping malls, and hospitals. Therefore, there is spatial correlation in traffic flow, which is the most obvious difference between traffic flow data and other time series data.

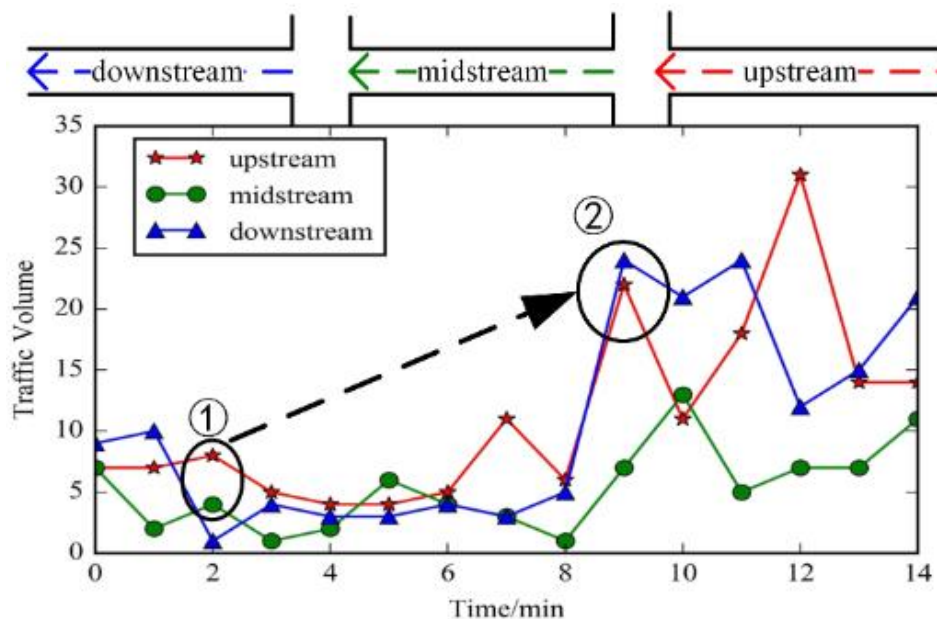


Fig. 3-2 Spatial dependence of traffic flow

3.1.3 Problem Definition.

In the context of traffic forecasting, two primary factors are considered: the configuration of the whole road network and data pertaining to traffic flow. This article presents the construction of a road topological network, whereby streets are represented as nodes. The experiment section utilizes traffic speed as an illustrative example of traffic information.

- **Definition 1:** Road Network G . As shown in Figure 2-3, the topological structure of an urban street network is constructed as an unweighted graph $G = (V, E)$, where V is a set of street nodes, $V = \{v_1, \dots, v_n\}$, N is the number of the nodes, and E is a set of edges. I use an adjacency matrix A to represent the connection between roads, A is in $R^{N \times N}$, where rows and columns in the dataset are assigned indices based on street sections. Each item in the dataset represents the level of connectedness between the associated street sections. The adjacency matrix only consists of binary elements, namely 0s and 1s. The value of the element is 0 in cases where there is no connection between streets, while a value of 1 indicates the presence of a link..

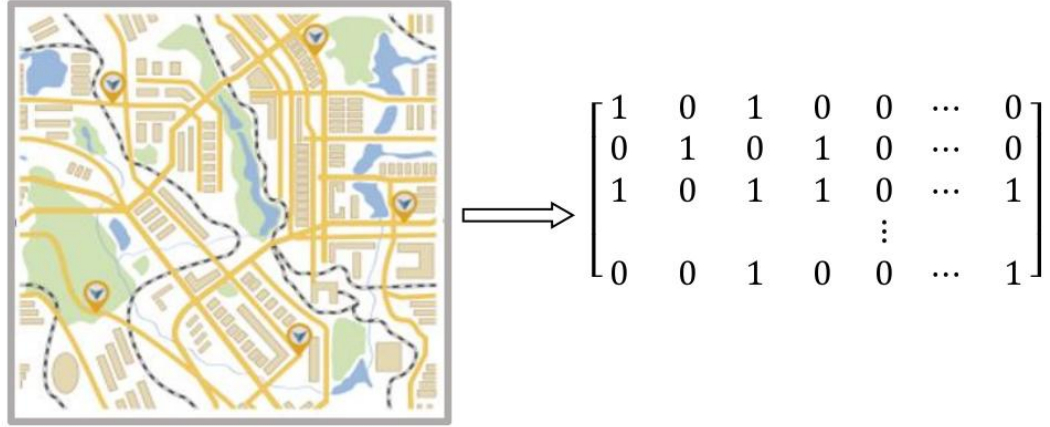


Fig. 3-3 Adjacency matrix of traffic road network

- **Definition 2:** Traffic flow feature matrix $X^{N \times P}$. The attribute feature of a node in the road network, specifically pertaining to traffic information such as traffic speed and bus number, is considered. In this context, P denotes the number of node attribute features, which corresponds to the length of the historical time series and $X_t \in R^{N \times i}$ is used to represent the speed on each road at the time i .

Therefore, the process of predicting traffic patterns by considering both temporal and geographical relationships may be described as the acquisition of a mapping function. This function is established using the road network G and the feature matrix X , which represents the characteristics of the road network. The equation used to compute the traffic speed of future T moments is represented as Equation 3-1:

$$[X_1, X_2, \dots, X_t] = f(G; (X_{t-n}, \dots, X_t)), \quad (3-1)$$

where n is the length of historical time series and T is the length of the time series needed to be predicted.

3.2 Framework

In this part, I introduce the framework of TC-GAT. Generally, this model consists of three parts: GAT, TCN, and Multi-head attention module. First, to improve the accuracy, the GAT is employed to extract spatial dependency by capturing the topological structure of the urban street network. Afterward, to improve the speed and extract the long-time series more effectively, the TCN is adopted to record the dynamic fluctuation of vehicle density on streets in order to establish temporal dependency. To tune up the importance of different time points, a Multi-head attention module is introduced. Finally, I use two fully connected layers to get results. Figure 3-4 shows the entire structure of my model.

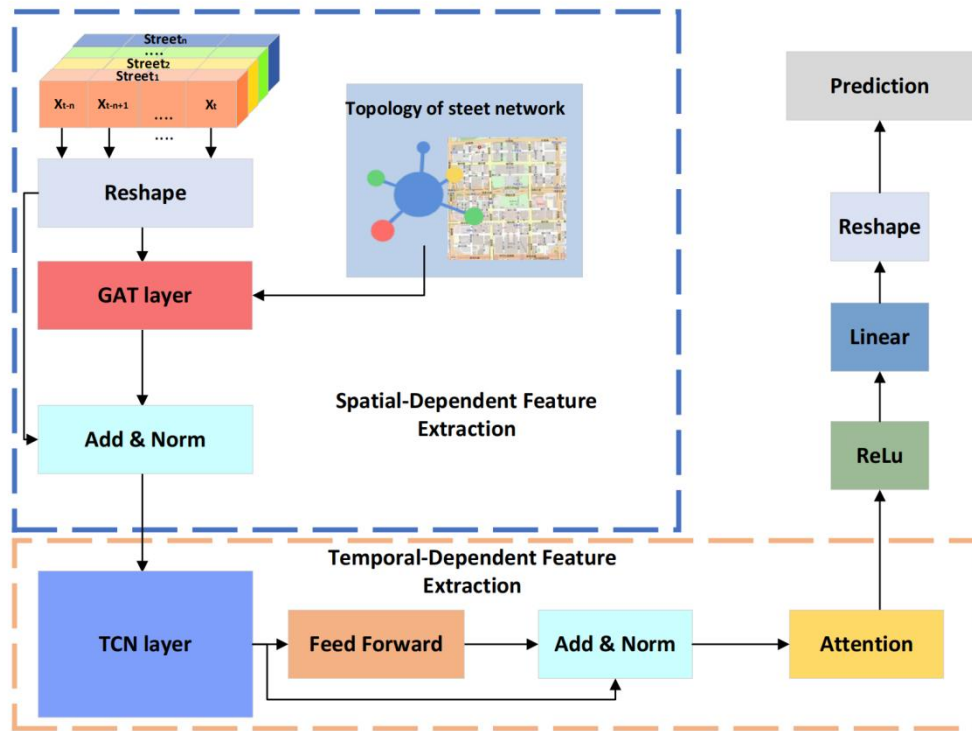


Figure 3-4: Structure of TC-GAT

3.2.1 Spatial-Dependent Feature Extraction Module

It is evident that conventional Convolutional Neural Networks (CNNs) are limited in their applicability to Euclidean domains, namely in the context of pictures and regular grids. The urban street network has a graph-like structure rather than a two-dimensional grid, rendering the CNN model inadequate in representing the intricate topological characteristics of the urban street network. Consequently, the CNN model fails to effectively describe spatial dependency. The Graph Convolutional Network (GCN) [52] represents a notable advancement in the field of Convolutional Neural Networks (CNNs), since it extends the use of CNNs to graph domains. The GCN model utilizes a Fourier domain filter to operate on the nodes of a graph and their immediate neighbors, enabling the extraction of spatial characteristics among the nodes. Subsequently, the GCN model may be constructed by sequentially stacking numerous convolutional layers. At now, GCN-based methodologies are regarded as very

sophisticated strategies in the field of traffic forecast research.

While GCN models have shown exceptional predictive accuracy, a common limitation is their tendency to capture static spatial dependencies in traffic, disregarding the potential evolution of these dependencies over time. In order to address this issue, the Graph Attention Network (GAT) is used in my model as an alternative to the Graph Convolutional Network (GCN) for the purpose of extracting spatial information from traffic data. The Graph Attention Network (GAT) enhances the Graph Convolutional Network (GCN) by integrating a distinct attention mechanism. Figure 3-5 illustrates the dynamic extraction of topological relationships using GAT, where node i represents the center street. This process involves encoding the topology of the street network and the characteristics on the roadways to capture spatial dependencies between the core street and its nearby streets.

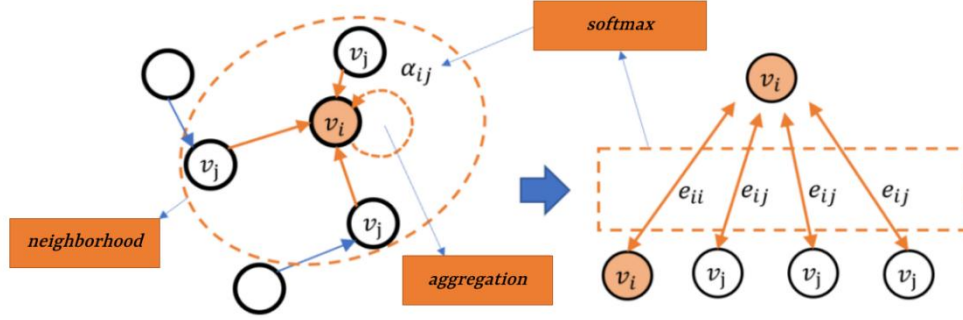


Figure 3-5: GAT mechanism

Specifically, to learn traffic as a graph, I first divide the urban area into streets. Assuming that the input to the GAT layer is a set of reshaped street features, $f = \{\tilde{f}_1, \dots, \tilde{f}_n\}$, $f \in \mathbb{R}^F$, where N is the number of streets, F represents the dimension of features in each street, the layer generates a new set of street features. The attention coefficients which indicate the importance of adjacent street j 's features to street i can be expressed as:

$$\alpha_{ij} = \frac{\exp\left(\text{LeakyReLU}\left(\vec{a}^T \left[\mathbf{W}\vec{f}_i \parallel \mathbf{W}\vec{f}_j\right]\right)\right)}{\sum_{k \in \mathcal{N}_i} \exp\left(\text{LeakyReLU}\left(\vec{a}^T \left[\mathbf{W}\vec{f}_i \parallel \mathbf{W}\vec{f}_k\right]\right)\right)}, \quad (3-2)$$

where $\mathbf{W}$ represents the weight matrix, $\mathbf{a}$ represents a shared attentional mechanism which is a single-layer feed-forward neural network, parameterized by a weight vector $\tilde{\mathbf{a}}$. T denotes transposition, $\parallel$ is the concatenation operation, and LeakyReLU represents the activation function.

The output feature of street i after aggregating the features of its adjacent streets is illustrated as:

$$\vec{d}_i = \sigma \left(\sum_{j \in \mathcal{N}_i} \alpha_{ij} \mathbf{W}\vec{f}_j \right), \quad (3-3)$$

where σ denotes a non-linearity.

In the end, I adopt a residual connection [39] followed by layer normalization. Hence, the output of the spatial-dependent feature extraction module is:

$$\text{LayerNorm}(X_t + P(X_t, A)), \quad (3-4)$$

where X_t is the feature matrix of street features at time t , A is the adjacency matrix of street topology, and $P(X_t, A)$ stands for the graph attention process explained by Equation (3-3).

3.2.2 Temporal-Dependent Feature Extraction Module

The present traffic speed is impacted by its prior states, indicating that predicting traffic speed involves analyzing time series data. Presently, the prevailing methodology involves the use of recurrent neural networks (RNNs) such as long short-term memory (LSTM) and gated recurrent unit (GRU) for the analysis of time series data. The recent concept of TCN demonstrates that a basic convolutional network outperforms the recurrent neural network (RNN) in a diverse range of tasks. In the job of copying memory, the Temporal Convolutional Network (TCN) demonstrates a perfect accuracy rate of 100% across all sequence lengths. However, the Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) models deteriorate to random guessing as the sequence lengths increase. In contrast, the parallel execution of convolutions is facilitated by the use of the same filter across several layers, resulting in enhanced computational efficiency of Temporal Convolutional Networks (TCN) as compared to Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) models. Hence, the decision has been made to use Temporal Convolutional Networks (TCN) as opposed to Recurrent Neural Networks (RNN) for the purpose of extracting time-dependent characteristics.

TCN proposes a hierarchical structure based on a 1D fully-convolutional network (FCN) architecture and causal convolutions. 1D FCN is used to make sure that the network produces an output of the same length as the input. The function of causal convolutions is to ensure no leakage from the future into the past. Figure 3-6 shows an example of the temporal convolutional unit with dilated factors $d = 1, 2, 4$ and filter size $q = 3$.

Since convolutions can be done in parallel, TCN is faster than the RNN-based method. For every input within the past flow sequence to be extracted by the filter, we can choose a large filter size k and step up the dilation factor d . The number of the extracted input of one such layer can be expressed as $(k-1)d$, which is capable of exponential growth d times, so it is more effective when facing long sequential data.

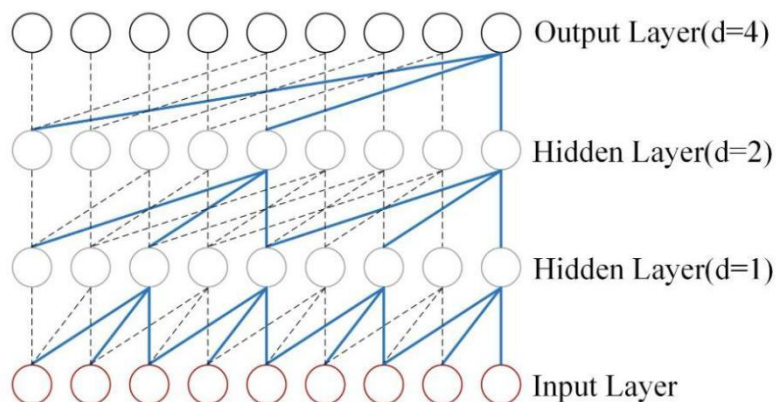


Figure 3-6: An example of Temporal Convolutional Unit.

The output of the spatial-dependent feature extraction module will become the input of the temporal-dependent feature extraction module as the hidden state h_t , assuming that the TCN has k layers, the output of the k^{th} layer can be expressed as:

$$h_t^{i,k} = \text{ReLU} \left(\text{Conv}_{1 \times 1} \left(h_t^{i,k-1} \right) + f_{dr} \left(h_t^{i,k-1} \right) \right), \quad (3-5)$$

where h_t^i represents the hidden state of the t^{th} time in the k^{th} layer for a specific region i , ReLU is an activation function, f_{dr} represents a dilated residual block that embodies dropout and ReLU.

Then, to extract deeper features and improve prediction accuracy, a fully connected feed-forward network consisting of two linear transformations with a ReLU activation in between is adopted to handle the output of TCN. A residual connection followed by layer normalization is also adopted in the end. Hence, the output of temporal-dependent feature extraction module is:

$$\text{LayerNorm}(h_t^{i,k} + \max(0, h_t^{i,k} W_1 + b_1) W_2 + b_2). \quad (3-6)$$

3.2.3 Attention mechanism layer

Google suggested the use of the self-attention mechanism as a replacement for the conventional recurrent neural network inside the transformer architecture. This approach has shown effective applications in the domain of natural language processing, and subsequently, it has been extended to other domains such as temporal prediction and image processing. Utilizing the benefits of the self-attention mechanism in long-term forecasting, it is a viable avenue of study to apply this approach to the prediction of traffic flow.

Given the extensive temporal scope and substantial scale of traffic flow data, actually, TCN remains a CNN-based method. While TCN exhibits improved efficacy in handling lengthy temporal sequences in comparison to Recurrent Neural Networks (RNN), it still encounters the issue of "forgetting". Notably, the forgetting phenomenon becomes more pronounced as the sequences grow longer. The TCN system has limitations in effectively capturing crucial traffic flow data features due to the continual influx of data. Consequently, the prediction process is compromised as it fails to retain vital feature information, resulting in a negative impact on the accuracy and efficiency of predictions. To address the limitation of the TCN model in retaining information from distant past, the integration of the attention mechanism enhances the model's ability to capture intricate mappings and dependencies among traffic flow data patterns. This enables the model to prioritize and emphasize crucial data features.

The attention mechanism is utilized in the prediction model to assign varying weights to the features of traffic flow data. The weights are larger for more significant feature information, allowing the model to disregard less important information. This enables the model to effectively capture fluctuations in traffic flow data by focusing on the important feature information that has a high impact on the predicted value. Additionally, the model utilizes the high-impact traffic flow information to perform traffic flow prediction, updating the header parameter promptly. By correctly capturing the long-term interdependence in the traffic flow sequences, the model enhances its prediction accuracy.

First, the attention mechanism consists of three main components: Query Q : an attention computation is performed to get the target output value; Key K : the context vector corresponding to the value; Value V : the selected input vector. The process of calculating Attention value in this work is illustrated in Fig. 3-7:

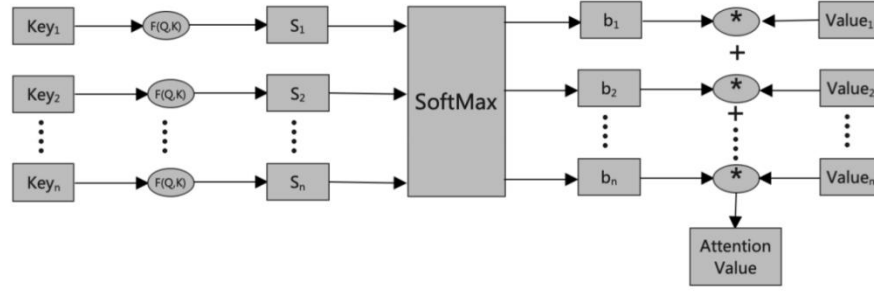


Figure 3-7: The process of calculating Attention Value.

First of all, according to *Query* and *Key_i* for correlation calculation to get the weights, generally use the vector dot product, c similarity and add additional neural network to find the value of three methods, the specific formula as shown in (3-7), (3-8) and (3-9), and the similarity between a and b is represented by d, and the range of the weights is also different using different methods.

$$S(\text{Query}, \text{Key}_i) = \text{Query} \cdot \text{Key}_i \quad (3-7)$$

$$S(\text{Query}, \text{Key}_i) = \frac{\text{Query} \cdot \text{Key}_i}{\|\text{Query}\| \cdot \|\text{Key}_i\|} \quad (3-8)$$

$$S(\text{Query}, \text{Key}_i) = \text{MLP}(\text{Query} \cdot \text{Key}_i) \quad (3-9)$$

Next, the normalized weights are obtained to the corresponding weight coefficients, and the weights of the first step are numerically converted by the introduced *Softmax* calculation method, which is mainly for normalization and highlighting the important features in the traffic flow data, and the specific calculation formula is shown in Equation (3-10):

$$b_i = \frac{\exp(S(\text{Query}, \text{Key}_i))}{\sum_{j=1}^m \exp(S(\text{Query}, \text{Key}_j))} \quad (3-10)$$

Finally, the weight coefficients corresponding to the weights *b_i* and *Value* obtained in the second step are weighted and summed to obtain the *Attention value*. The calculation formula is shown in Equation (3-11):

$$\text{Attention}(\text{Query}, \text{Source}) = \sum_{i=1}^{L_x} b_i \cdot \text{Value}_i \quad (3-11)$$

The significance of joining the attention mechanism is reflected in the following three aspects: First of all, by adding the attention mechanism, the model can obtain the ability of "focusing", and its weight matrix assigns corresponding weight values to different features according to the degree of influence on the traffic flow, and these weight values are multiplied with the output vector of the TCN to calculate the product, and the values obtained from the product calculation are obtained by the weighted average operation to get the value of the weight matrix. The values obtained from the product calculation are weighted and averaged to obtain the values of the weight matrix. Secondly, by utilizing the attention mechanism to selectively memorize the feature information, the model can capture the long-term dependency relationship well when facing the problem of long time series. Moreover, the Attention

mechanism can be used to extract potential mappings between two completely different modalities and assign weights to them for different relationships, by adding Attention, the potential and complex mappings and dependencies between modalities of traffic flow data can be better captured.

Therefore, this paper incorporates the Attention Importance mechanism into the traffic flow prediction model to obtain better prediction results, and find the important factors affecting the fluctuation of traffic flow by calculating the Attention Importance Score of the traffic data and its features and assigning weights to them.

CHAPTER 4: Experimental Design

4.1 Data Description

Accurate traffic flow data collection is crucial for traffic flow prediction, the source and quality of the data are factors to be considered, and a complete and accurate traffic flow dataset can maximize the accuracy of the prediction. Early research in the field of transportation was mainly conducted by manual data collection, but this method has too many disadvantages, such as low accuracy of the collected data, high labor consumption and low efficiency, affecting the normal operation of the road traffic, and the personal safety of the collection personnel. With the gradual maturity of data collection technology, sensors and other collection devices are gradually used instead of manual labor for automatic real-time traffic flow data collection. The collected data are transmitted to the traffic data management center in various ways, and after data cleaning and processing, they are organized and summarized and opened to the public, giving researchers in the field of transportation great data support.

In this part, an evaluation is conducted to assess the prediction performance of the TC-GAT model on two real-world datasets, namely the SZ-taxi dataset and the Los-loop set. Both of these datasets are linked with the variable of traffic speed. To ensure the applicability of the findings, traffic speed is selected as the representative traffic information in the experiment section, without sacrificing the generalizability of the results.

- **SZ-taxi:** This dataset represented Shenzhen's taxi trajectory from January 1 to January 31, 2015. Luohu District's 156 key roadways are selected as the study region. We chose the Luohu District's 156 key roadways as the study region. One illustrates the spatial relationship between roadways using a 156*156 adjacency matrix. Each row of the matrix represents a distinct road, whereas the values inside the matrix indicate the level of connectedness between these roads. Another component is a feature matrix that provides a comprehensive overview of the temporal variations in speed for each route. In this representation, each row corresponds to a distinct route, while each column denotes the traffic speed observed on these roads at various time intervals. The velocity of traffic on each roadway is consolidated at 15-minute intervals.
- **Los-loop:** This dataset was gathered in real-time by loop detectors on a Los Angeles County highway. From March 1 to March 7, 2012, 207 sensors and their traffic speed were chosen. Every 5 minutes, we summed up the traffic speed. Similarly, the data provides an adjacency matrix and a feature matrix. The distance between sensors in the traffic networks is used to calculate the adjacency matrix. The linear interpolation method is employed to fill in the missing values.

4.2 Data Processing

Due to aging sensors, human factors, and weather conditions, the traffic flow data collected on many roads may be abnormal, missing, or duplicated. If the original traffic flow dataset is used directly, it will cause deviation in the prediction of traffic flow data, which will lead to deviation in road management and related resource costs, resulting in unnecessary waste. Therefore, data preprocessing is a necessary process for traffic flow prediction.

For the abnormal, missing and duplicate problems in the original data, the corresponding methods are used to detect and repair to ensure the completeness and accuracy of the predicted sample data, which is also of great significance to improve the accuracy of the traffic flow prediction model.

The traffic flow dataset used in this paper is the information directly collected by the detection site, which contains a large amount of noise information and missing information, which will affect the prediction results of the model if not processed, and the model prediction will be unstable, inaccurate or even erroneous. Therefore, according to the problems existing in the datasets used in this paper, it is necessary to preprocess the two datasets. The preprocessing process for these traffic datasets consists of three main aspects: data repair, denoising and normalization.

First, data repair. Traffic data contains two kinds of anomalies: the first is missing data, due to the failure of traffic sensor equipment, transmission interruption between the sensor and the system, etc., resulting in a longer period of time without obtaining any data; the second is the data errors, such as the appearance of negative values and far beyond the historical values of the same period of time to take the value, single-value repetition of the anomalies, and so on. For the above two kinds of abnormal data, appropriate methods are selected to repair them. For the missing data, the traditional methods include historical mean estimation method, lane ratio method and linear interpolation method. In this paper, the historical mean estimation method is chosen to fix the missing data. As for the error data, it is first recognized and then deleted, and then the data is made up as in the case of missing data.

Second, denoising. When the traffic sensor collects data, if the sampling interval is short it is easy to appear Gaussian noise, noisy data will directly affect the prediction effect of the model, so it is necessary to reduce the noise in the data. Common noise reduction methods include wavelet algorithms, exponential smoothing and Kalman filtering. In this paper, the exponential smoothing method is selected for denoising.

Third, normalization. When modeling traffic data using deep learning models, the data needs to be normalized. Normalization removes the effect of the magnitude of the data and improves the accuracy of the model. In this paper, Z-score standardization is used to process the data so that the mean of the data is 0, the standard deviation is 1, and the relative size and trend of the data do not change, which is convenient for comprehensive analysis.

4.2 Evaluation Metrics

The evaluation index of the model involves comparing the projected value with the actual value from several perspectives. This allows for a quantitative analysis of the discrepancy between the two values, enabling a comparison of the model's performance differences. In the domain of traffic flow prediction, the level of agreement between the predicted value and the actual value is directly associated with the accuracy of the prediction model. A higher level of agreement indicates a higher accuracy in the model's predictions, while a lower level of agreement suggests a poorer predictive capability. To assess the performance of the

TC-GAT model, many widely used assessment metrics are used.

- **Root Mean Square Error (RMSE).** The calculation involves taking the average of the squared differences between the anticipated values generated by the model and the actual values. The resulting value falls within the range of $[0, +\infty)$. When the anticipated value aligns perfectly with the actual value, its discrepancy is zero. A model's prediction effectiveness is shown by the proximity of its root mean square error (RMSE) value to zero. A smaller RMSE value suggests a stronger prediction effect, whereas a larger RMSE value indicates a worse prediction effect and a higher level of error. The evaluation of data changes may be facilitated to some degree by using the Mean Squared Error (MSE) metric. As the root mean square error (RMSE) approaches zero, the predictive model's capacity to accurately depict the experimental data strengthens, resulting in a greater level of model accuracy [49].

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (Y_t - \hat{Y}_t)^2} \quad (4-1)$$

- **Mean Absolute Error (MAE).** When the Mean Absolute Error (MAE) number is zero, it indicates that the model's prediction performance is flawless. When comparing the root mean squared error (RMSE) with the mean absolute error (MAE), it can be seen that MAE normalizes the error of each data point, hence mitigating the impact of absolute errors resulting from specific outliers. However, it is important to acknowledge that in cases when the actual value contains a data point of 0, there arises a potential issue where the denominator may also be 0, rendering this formula inapplicable. Considering the fact that Mean Absolute Error (MAE) has the ability to provide a more comprehensible representation of the error between predicted and true values, this article has selected MAE as one of the assessment metrics for the model [50].

$$MAE = \frac{1}{\xi} \sum_{j=1}^{\xi} \sum_{i=1}^N |Y_{t+1,j}^i - \hat{Y}_{t+1,j}^i| \quad (4-2)$$

4.3 Loss Function

The time series of traffic flow, as recorded by sensors, is subject to several major inaccuracies. The inclusion of these substantial erroneous data points contributes to an increase in the forecast error associated with the traffic flow series. In conventional deep learning architectures, the mean square error is often used as the loss function to facilitate gradient updating. However, this approach is susceptible to overfitting in scenarios when the input data exhibits a substantial amount of noise. To mitigate the influence of noisy data on the gradient update during the training of the model, an enhanced loss function is used in this model. The formulation of the loss function is presented as follows:

$$\rho(e) = \frac{\frac{1}{m} \sum_{i=1}^L (Y_i - X_i)^2}{2 + \alpha \sqrt{\frac{1}{L} \sum_{i=1}^L (Y_i - X_i)^2}}, \quad (4-3)$$

where Y and X are the predicted and true values of the model, respectively, L is the step size, e is the predictive error value. The larger the value of a , the greater the suppression of the loss function on the gradient updating due to the measurement error.

For different values of a , the value of this derivative function varies with the error e as shown in

Figure 4-1:

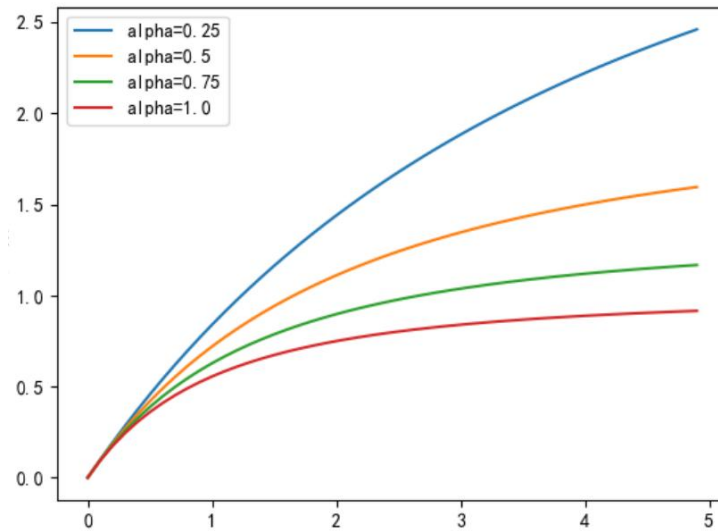


Figure 4-1: Variation curve of influence function with measurement error.

It can be seen that the value of the derivative function of the loss function converges to a constant value with the increasing error between the real data and the error data, which indicates that when the measurement error increases infinitely, the value of the loss used for the gradient backpropagation will not increase rapidly, thus suppressing the impact of the measurement error on the model training. Moreover, the second-order derivative of the loss function is always positive, so the loss function is a convex function, which is conducive to the iterative learning of the learnable parameters in the deep learning model, and effectively avoids the model falling into the local minima in the gradient descent process.

4.4 Baselines

In this study, the performance of the TC-GAT model is compared with many baseline approaches.

- **The History Average model (HA)** is a statistical model used in the field of history. The historical average model utilizes average values from preceding periods to forecast future flow levels. The inputs consist of the data obtained from the local flow.
- **The Autoregressive Integrated Moving Average (ARIMA)** model is a commonly used statistical model in time series analysis. The autoregressive integrated moving average (ARIMA) is a well recognized technique for forecasting time series data. It incorporates both autoregressive and moving average components to effectively model and analyze time series patterns.
- **The Support Vector Regression model (SVR)** is a machine learning algorithm that is used for regression tasks. The model is trained with past data in order to ascertain the correlation between input and output variables. Subsequently, it use future traffic data to make predictions. The linear kernel is used, with a penalty term of 0.001.
- **The Gated Recurrent Unit (GRU)** is a kind of recurrent neural network (RNN) architecture. The proposed methodology obtains the current traffic status at time t by including the latent status at time $t-1$ and the present traffic data as input variables. The model is capable of capturing temporal dependence by including the changing trend of earlier traffic data while recording current traffic data.

- **The Temporal Convolutional Networks model (TCN)** is a computational framework that is used for modeling temporal data. The selected model for this study utilizes a filter size of 3 and a dilation factor of 2.
- **The Temporal Graph Convolutional Network (T-GCN)** is a computational model that operates on temporal graphs. The use of the GCN and GRU models is employed to encode geographical and temporal information, respectively, in order to enhance the performance of traffic flow prediction tasks.

I also analyze the effects of the various components presented in our model. The following two alternative models are offered to validate distinct sections of my design.

- **TCN+GAT.** In this structure, I use only GAT and TCN to extract temporal and spatial features respectively, without combining the Attention mechanism to adjust the importance of different time points.
- **TCN+Attention.** This structure combines TCN and Attention mechanism to extract temporal feature but ignores spatial feature.

I utilize the identical testing data for all models and replicate the training data for each model separately to ensure fair comparisons. Furthermore, all models employ the same loss function as this model.

4.6 Experimental settings

The experimental equipment for this paper is a server with Intel Core i9-9900X processor, 128G RAM, NVIDIA GeForce RTX 3090 graphics card, running Linux, and python3.9 development environment. On this equipment, the algorithms of this chapter and various baseline models are implemented respectively.

The models in this paper are built using Pytorch+Keras and some commonly used Python libraries. In this paper, the batch_size is set to 256, the Epoch size is 500, and the Dropout loss rate is 20%. Dropout can effectively prevent overfitting, and the key idea is to temporarily discard a portion of the intermediate layer units from the network according to a certain probability during network training, which provides a way to efficiently combine many different It provides a way to effectively combine many different neural network structures [48], and is a way to add noise to the hidden units of the neural network to regularize the neural network, which can effectively improve the generalization ability of the model.

The hyperparameter α in the loss function determines the ability to suppress the measurement outliers during the model training process; too small a value of α makes the outliers have a greater impact on the model training, whereas too large a value of α reduces the training speed of the model and introduces the problem of oversmoothing. Therefore, the model prediction accuracy is compared here for different values of the parameter α of the improved loss function. After calculation, relative to the case of $\alpha = 1$, the prediction accuracy of the model decreases with the increase of the value of α , and the prediction accuracy of the model is gradually lower than that of the model trained with the standard loss function, and when the value of α decreases, the prediction accuracy of the model also decreases but its prediction effect is better than that of the standard loss function. The prediction effect of the trained model, it can be seen that 1 is a more appropriate value for α .

CHAPTER 5: Results & Discussions

Traffic flow data show great regularity in a certain time period, but there are differences in different time periods. On the one hand, the fluctuation of traffic flow on the same road section is similarly affected by people's life pattern, and the traffic flow between road sections in the city is correlated, and the historical traffic condition of adjacent road sections also affects the traffic condition of future road time periods. On the other hand, there are differences in the form and characteristics of traffic flow on different roads in the road network and at different time points or monitoring points on the same road. In general, traffic flow data are characterized by large dimensions, many features, and large computation volume. For these complex multiple features in traffic flow data, this paper first uses GAT to extract the spatial features of the data to achieve the effect of compression of data, which can reduce the complexity of the traffic prediction calculation and improve the operational efficiency; then the output of GAT is used as the input of TCN, which can effectively solve the time-dependence problem in the traffic flow data, and finally Attention is used to assign weights to the multiple features of the traffic flow data. Multiple features of the data are given weights to find the "important influencing factors" of the traffic flow fluctuations, so as to improve the accuracy of traffic flow prediction. Table 5-1 illustrates our prediction model and alternative baseline approaches on the SZ-taxi and Los-loop datasets for 15 and 30 minutes. It can be demonstrated that our model achieves the best prediction performance across all evaluation metrics and prediction horizons, demonstrating the model's usefulness in spatio-temporal traffic forecasting. Meanwhile, this model not only can effectively reflect the general trend of traffic flow changes in a day, and the predicted values are smoother than the actual values of the data, but also can effectively fit the cyclical characteristics of the traffic flow data, and its predicted values overlap well with the actual values.

Method	SZ-taxi				Los-loop			
	15min		30min		15min		30min	
	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE
HA	7.9198	5.4969	7.9198	5.4969	7.4427	4.0145	7.4427	4.0145
ARIMA	8.2151	6.2192	8.2123	6.2144	10.0439	7.6832	9.3450	7.6891
SVR	7.5368	4.9269	7.4747	4.9819	6.0084	3.7285	6.9588	3.7248
GRU	4.0483	2.6814	4.0769	2.7009	5.2182	3.0602	6.2802	3.6505
TCN	4.0021	2.6955	4.0114	2.6530	5.1056	3.0079	6.1265	3.6428
T-GCN	3.9162	2.7061	3.9617	2.7452	5.1264	3.1802	6.0598	3.7466
TCN+GAT	3.8451	2.6513	3.9024	2.6378	5.0894	2.9877	5.9685	3.6112
TCN+ATT	3.9755	2.6654	3.9987	2.6934	5.0986	2.9986	6.0053	3.6357
TC-GAT	3.7487	2.5865	3.8857	2.6015	4.9852	2.8657	5.9177	3.5981

Table 5-1: Comparisons with Different Baselines

(1) High predictive accuracy. It is evident that neural network-based methodologies, which prioritize the modeling of temporal characteristics, exhibit superior predictive accuracy compared to other baseline models, including the HA model, the ARIMA model, and the SVR model. The primary reason for this issue is from the challenges faced by existing methodologies in effectively handling complex, nonstationary time series data. These methodologies mostly depend on numerical aspects while disregarding the spatial and temporal correlations present in the data. Moreover, in terms of traffic forecasting, the ARIMA method is considered to be less accurate compared to the HA method due to its limitations in handling long-term and nonstationary data. The ARIMA model calculates the error of each data point and takes the average, which can lead to increased total error if there are fluctuations in certain data points. Furthermore,

the variability of traffic flow is influenced by individuals' lifestyles, resulting in cyclical fluctuations. The overall fluctuation of traffic flow is substantial, and there is a notable disparity between traffic flow fluctuations on weekdays and weekends. However, the predictive outcomes of the proposed method align closely with the actual traffic flow patterns during peak and off-peak periods. Consequently, the method accurately forecasts the overall traffic flow trend, irrespective of whether it is a weekday or a weekend.

(2) Spatio-temporal prediction capability. To evaluate whether our model can depict spatial and temporal features from traffic data, we compare it to the TCN+Attention model. We can obviously observe that the model based on spatiotemporal characteristics (TC-GAT) has higher prediction precision than those based on a single component (TCN+Attention), demonstrating that our model can extract spatial and temporal information from traffic data. It can be seen that my model can roughly predict where the overall traffic is going. The RMSE and MAE of our model are reduced by roughly 3.3% and 2.9%, respectively, when compared to the TCN + Attention model, which analyzes just temporal aspects, demonstrating that our model can capture temporal dependency well.

(3) The efficacy of capturing the significance of various temporal intervals. In order to evaluate the efficacy of capturing the significance of various time points, a comparison was conducted between our model and the TCN+GAT model. The results indicate that our model exhibits a decrease in RMSE of around 0.86% and an increase in accuracy of approximately 0.32% compared to the TCN+GAT model when applied to a 15-minute time series. The use of the Attention mechanism allows for the dynamic adjustment of the contribution degree at various time points. Therefore, the accuracy of the forecast is enhanced. Furthermore, the proposed model demonstrates the capability to effectively predict local mutation data. This suggests that the model not only successfully extracts temporal correlation and periodicity characteristics in traffic flow, but also identifies significant factors influencing traffic flow fluctuations. Consequently, the model is also proficient in predicting mutation flow.

(4) The efficacy of Temporal Convolution Networks (TCNs). Based on the given input data, it is seen that the training time for our model is 30 minutes, but the typical GRU-based model requires 50 minutes. This comparison suggests that the TCN exhibits more efficiency in comparison to the GRU model. The hierarchical structure of TCN enables the analysis of data in distinct time periods, facilitating the efficient generation of forecasts by leveraging extensive previous data. In contrast to the GRU model, which necessitates sequential prediction, the TCN is capable of effectively processing extended input sequences due to the use of identical filters in each layer. Additionally, the findings of this study demonstrate that the model used in this research not only enhances prediction accuracy, but also successfully enhances the model's fitting performance.

CHAPTER 6: Conclusions and Recommendations

Through the research and analysis of previous research scholars in the field of transportation and the challenges and techniques of existing traffic flow prediction research, it is found that traffic flow data is different from general time series data, and the change of traffic flow data is affected by people's living habits, weather conditions, personal vehicle speed, road network structure and other multiple factors, but it is difficult to take these factors into account in the actual research. However, in actual research, it is difficult to take these factors into account in the research method, and the traditional traffic flow prediction model assumes that the traffic flow sequence contains only linear correlation structure, and it is difficult to deal with non-linear correlation of traffic flow data effectively, and the feature extraction of the multilayer

non-linear mapping principle of the deep learning model can effectively obtain the high-dimensional non-linear correlation features of traffic flow data, and the use of deep learning techniques and theories to build scientific and efficient traffic estimation and prediction system can effectively reduce traffic congestion. The use of deep learning technology and theory to construct a scientific and efficient traffic estimation and prediction system can effectively reduce traffic congestion and improve road capacity and utilization. Therefore, in this paper, I construct a traffic flow prediction model by using deep learning models and techniques such as GAT, TCN and attention mechanism, and the main results of this work are summarized as follows:

(5) This research status and research history of traffic flow prediction algorithms at home and abroad are described, the necessity of traffic flow prediction, the prediction principle are given, the shortcomings of the existing traffic flow prediction methods and the direction to be improved are described in detail, and a good foundation is laid for the subsequent proposal of new prediction models.

(6) According to the problems encountered in previous research on traffic flow prediction, a fusion model based on GAT-TCN is constructed to predict traffic flow, in which GAT is mainly responsible for data spatial feature extraction and TCN layer is responsible for data prediction, and the experimental results show that the prediction results of the model can follow the general trend of the actual values very well, but for some of the peaks and valleys of the traffic flow can not be predicted very well, although compared with the separate TCN model, the prediction accuracy is improved considerably, but the improvement in the performance of its fitting is not obvious.

(7) According to the problems encountered in the GAT-TCN fusion model, the TC-GAT model based on the attention mechanism is constructed by adding the Attention layer to the model to obtain the traffic flow information that has a high influence on the prediction value. The model first uses GAT to obtain the high-dimensional potential spatial features in the multi-mixed features in the traffic flow data, and reduces the dimensionality to achieve the purpose of data compression, which can effectively reduce the workload of the traffic flow prediction network at a later stage; and then it uses the TCN network module to extract the time-dependent features in the traffic flow data, which reduces the influence of time-dependent features on the prediction results and effectively improves the accuracy of the traffic flow prediction. The TCN network module is used to extract the time-dependent features in the traffic flow data to reduce the influence of time-dependent features on the prediction results and effectively improve the accuracy of the traffic flow prediction. The experimental results show that compared with the original model, the TC-GAT model based on the attention mechanism has improved both the error and fitting performance.

(8) SZ-taxi and Los-loop are selected as two public datasets for experiments. Due to the different road structures of the two datasets, the fluctuations of their traffic flow data are also significantly different, and the tests on the two different datasets can better verify the applicability and effectiveness of the model. The experimental results show that the TC-GAT model based on the attention mechanism constructed in this paper can effectively capture the time dependence, periodicity and nonlinearity among the sequences, and has good prediction effect on both datasets. Finally, the model proposed in this paper is compared with the baseline model, and by comparing their prediction errors, it is concluded that the model proposed in this paper has better prediction performance. Meanwhile, the necessity of each module of the model, such as the graph attention network layer, the time coding module, and the attention mechanism, is verified through ablation experiments.

However, in this paper, two problems in the task of traffic flow prediction: spatio-temporal correlation modeling and the balance between prediction accuracy and time overhead have been studied in depth and certain results have been achieved, but there are still many deficiencies that need to be

investigated and solved in future work.

(1) The TC-GAT proposed in this paper only considers the spatio-temporal correlation of traffic flow data, and in real scenarios, many external factors also have a large impact on the traffic flow state. Meanwhile, TC-GAT constructs asynchronous spatio-temporal links based on spatial distances, ignoring the rich semantic information between nodes. In addition, due to the mechanism of constructing asynchronous spatio-temporal graphs in TC-GAT, its performance will be degraded when dealing with very large-scale traffic networks.

(2) The self-attention mechanism enhances the long-term prediction ability of the model, but increases the complexity of the model and the computational cost, adopting a certain sparse method to simplify the computation of the self-attention value so as to reduce the complexity of the model is a direction that needs to be studied in depth.

(3) The model in this paper is currently only applied to traffic flow prediction tasks, and has not been applied in other spatio-temporal tasks. However, the prediction model in this paper should be general and can be applied to any data prediction with spatio-temporal characteristics. And different tasks may have different graph structures, which is a challenge for GAT. So future research work should also aim to extend the application of the model.

(4) The traffic dataset used in this paper does not contain detailed information about the location of traffic nodes, weather, traffic accidents, and other factors affecting traffic flow, and these factors are not taken into account when constructing the model. Therefore, it will be helpful to collect more detailed traffic information and take it into account when constructing the model, which will help to improve the prediction performance of the model.

In view of the above shortcomings, this paper will next consider extracting more complex traffic flow feature information, comprehensively analyzing the impact of multiple factors on traffic flow and assigning corresponding weights to them for model training and prediction, and validating the model on multiple road datasets at home and abroad in various aspects. In addition, processing traffic flow big data and applying it to real-time traffic prediction research is of fundamental significance to the construction of new smart cities in the future, and it is an important part of the future transportation research field to realize a more timely presentation and application effect in the road by continuously improving the accuracy of the prediction model and updating it timely, accurately, and dynamically to the intelligent transportation system to be ultimately applied to the vehicular network, and distributing it to the vehicles and intersections of each road and other nodes. Transportation research field is an important content.

CHAPTER 7: REFLECTION

Because of this research, I was able to significantly expand both my theoretical understanding and my practical experience. This expertise encompasses not only the fundamental concepts of machine learning and deep learning, but also the domain of spatio-temporal data processing and the practice of data cleaning. The course taught me the fundamentals of machine learning, and Mr. Gordon's excellent explanation piqued my interest in applying machine learning to the solution of real-world issues. Because I want to focus my study on VANET in China, I choose to investigate an issue that is connected to traffic flow prediction. I first gained an understanding of the fundamental research content and the most recent research advances in the fields of traffic flow data mining and deep learning by reading relevant literature both in the United States and overseas. I have a greater grasp of the research history of this topic, as well as

its current status and relevance, as well as the current state-of-the-art approaches, as a result of reading the representative literature that has been published in this field in recent years. A tentative study proposal about this subject has been provided after conducting an analysis of the existing difficulties associated with traffic flow prediction and the drawbacks of the research approaches that are currently in use. The objective of the traffic flow spatio-temporal correlation mining job is to differentiate the levels of contribution made by various traffic nodes and time nodes to the overall traffic road network. This information can then be utilized to make more accurate predictions.

Throughout the process of finishing this topic, I picked up new skills, including the ability to see difficulties, think critically about them, find solutions to those problems, enhance my approaches, and cultivate the habit of actively gathering and looking for information. It was a really fascinating experience to apply the skills that were learnt to actual challenges in the real world. Last but not least, when I was working on my thesis, I was able to enhance not only my writing abilities in English but also my skills in inputting formulae and creating diagrams. In spite of the fact that I have almost finished the project in its entirety, there are still some issues that need to be fixed. In my effort on researching literature, I did not accomplish a job that was sufficiently methodical. I did not conduct an exhaustive investigation on the most recent work. I invested a significant amount of time on reading and summarizing a variety of works. Because of this, I am also aware that I need to increase the amount of material I read in order to enhance my ability to summarize. Throughout the course of the experiment, I have come up with a few ideas that I have not yet put into practice. On the one hand, I haven't learned the procedures for the realization very well, and on the other hand, the reproductions of the models haven't generated very satisfactory results. As a result, I want to devote more time and effort in the future to developing my coding skills. In addition, I didn't have as much contact with my mentor, so we didn't talk about my work as much and I didn't receive as much guidance from him. In the long run, one of my primary goals is to become more proficient in the English language so that I can have more natural and fluent conversations with individuals of other nationalities.

Learning has been both interesting and unforgettable during this past year. It has been a privilege to get education from all of these amazing instructors. My English language and general knowledge abilities have significantly increased thanks to the online courses I've taken, despite the fact that I did not attend classes in person in Swansea. As I worked through the numerous assignments for this class, I was able to enhance my skills in all areas, but my ability to write code and essays in particular stood out as particularly beneficial. Time is like the sea, everyone is trying to build a huge ship belonging to their own, and in the sea of time that will never return to travel, we have different speeds, different purposes, sometimes clear skies, sometimes encounter a whirlpool, sometimes encounter a storm, some people go with the wind, some people want to turn back the same way, and some people have accidents. Slowly, we found that once the ship began to move it will not stop because of a certain person, and the time will not be stopped by any one person, we have to build a huge ship that can carry the wind and two in a limited time, we have to spend more time in the pre-build the foundation, and only a strong and solid foundation can withstand the test of time. Last but not least, I will always remember those things that I was lost, entangled, and found difficult at the moment, as well as those classmates and friends who walked with me in the darkness, and those UWTSD teachers who gave me a lot of guidance in my life and academics at important moments. Maybe this time is only a small part of my long life, but your companionship and support will be with me throughout the rest of my journey. Last but not least, I wish you all the best of luck in your future life, sail away, and arrive at your ideal island soon.

REFERENCES

- [1] Mori U, Mendiburu A, Álvarez M, et al. A review of travel time estimation and forecasting for advanced traveller information systems[J]. *Transportmetrica A: Transport Science*, 2015, 11(2): 119-157.
- [2] Meng, M. (2012). Spatial and Temporal Characteristics for Congested Traffic on Urban Expressway. *Journal of Beijing University of Technology*, *Journal of Beijing University of Technology*.
- [3] Duan, Y., Lv, Y., Liu, Y.-L., & Wang, F.-Y. (2016). An efficient realization of deep learning for traffic data imputation. *Transportation Research Part C: Emerging Technologies*, 168–181.
- [4] Mehdi, H., Pooranian, Z., & Vinueza Naranjo, P. G. (2022). Cloud traffic prediction based on fuzzy ARIMA model with low dependence on historical data. *Transactions on Emerging Telecommunications Technologies*, 33(3).
- [5] Zhang, Y., & Hou, Z. (2018). Short Term Traffic Flow Prediction Based on Improved Support Vector Machine. *Journal of Applied Science and Engineering*, *Journal of Applied Science and Engineering*.
- [6] [6]Johansson, U., Boström, H., Löfström, T., & Linusson, H. (2014). Regression conformal prediction with random forests. *Machine Learning*, 97(1–2), 155–176.
- [7] Fu, R., Zhang, Z., & Li, L. (2016). Using LSTM and GRU neural network methods for traffic flow prediction. 2016 31st Youth Academic Annual Conference of Chinese Association of Automation (YAC). Presented at the 2016 31st Youth Academic Annual Conference of Chinese Association of Automation (YAC), Wuhan, Hubei Province, China.
- [8] Mao, J., Huang, H., Lu, W., Chen, Y., & Liu, L. (2022). Multi-precision traffic speed predictions via modified sequence to sequence model and spatial dependency evaluation method. *Applied Soft Computing*, 130, 109700.
- [9] Zhao, L., Song, Y., Zhang, C., Liu, Y., Wang, P., Lin, T., ... Li, H. (2020). T-GCN: A Temporal Graph Convolutional Network for Traffic Prediction. *IEEE Transactions on Intelligent Transportation Systems*, 3848–3858.
- [10] Liu, Z., & Zhou, J. (2020). Graph Attention Networks. In *Synthesis Lectures on Artificial Intelligence and Machine Learning*, *Introduction to Graph Neural Networks* (pp. 39–41).
- [11] Bai, S., Kolter, J. Z., & Koltun, V. (2018). An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling. *arXiv: Learning*, *arXiv: Learning*.
- [12] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, AidanN., ... Polosukhin, I. (2017). Attention Is All You Need. *Neural Information Processing Systems*, *Neural Information Processing Systems*.
- [13] Stephanedes, YorgosJ., Michalopoulos, P., & Plum, R. (1981). IMPROVED ESTIMATION OF TRAFFIC FLOW FOR REAL-TIME CONTROL (DISCUSSION AND CLOSURE). *Transportation Research Record*, *Transportation Research Record*.
- [14] Okutani, I., & Stephanedes, Y. J. (1984). Dynamic prediction of traffic volume through Kalman filtering theory. *Transportation Research Part B: Methodological*, 1 – 11.
- [15] Zheng W, Lee D H, Shi Q. Short-term freeway traffic flow prediction: Bayesian combined neural network approach[J]. *Journal of transportation engineering*, 2006, 132(2): 114-121.
- [16] Lin, G., Lin, A., & Gu, D. (2022). Using Support Vector Regression and K-Nearest Neighbors for short-term traffic flow prediction based on Maximal Information Coefficient. *Information Sciences*, 517 – 531.

- [17] Ma, X., Dai, Z., He, Z., Na, J., Wang, Y., & Wang, Y. (2017). Learning Traffic as Images: A Deep Convolutional Neural Network for Large-Scale Transportation Network Speed Prediction. Cornell University - arXiv, Cornell University - arXiv.
- [18] Seo, Y., Defferrard, M., Vandergheynst, P., & Bresson, X. (2018). Structured Sequence Modeling with Graph Convolutional Recurrent Networks. In *Neural Information Processing, Lecture Notes in Computer Science* (pp. 362 – 373).
- [19] Li, M., & Zhu, Z. (2022). Spatial-Temporal Fusion Graph Neural Networks for Traffic Flow Forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence*, 4189 – 4196.
- [20] Wu, Z., Pan, S., Long, G., Jiang, J., & Zhang, C. (2019). Graph WaveNet for Deep Spatial-Temporal Graph Modeling. *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*. Presented at the Twenty-Eighth International Joint Conference on Artificial Intelligence {IJCAI-19}, Macao, China.
- [21] Ren, Y., Zhao, D., Luo, D., Ma, H., & Duan, P. (2022). Global-Local Temporal Convolutional Network for Traffic Flow Prediction. *IEEE Transactions on Intelligent Transportation Systems*, 1578 – 1584.
- [22] Wu, K., Wu, J., Feng, L., Yang, B., Liang, R., Yang, S., & Zhao, R. (2021). An attention-based CNN-LSTM-BiLSTM model for short-term electric load forecasting in integrated energy system. *International Transactions on Electrical Energy Systems*, 31(1).
- [23] Guo, S., Lin, Y., Feng, N., Song, C., & Wan, H. (2019). Attention Based Spatial-Temporal Graph Convolutional Networks for Traffic Flow Forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence*, 922 – 929.
- [24] Guo, K., Hu, Y., Qian, Z., Sun, Y., Gao, J., & Yin, B. (2022). Dynamic Graph Convolution Network for Traffic Forecasting Based on Latent Network of Laplace Matrix Estimation. *IEEE Transactions on Intelligent Transportation Systems*, 1009 – 1018.
- [25] Xu, C., Feng, J., Zhao, P., Zhuang, F., Wang, D., Liu, Y., & S. Sheng, V. (2021). Long- and short-term self-attention network for sequential recommendation. *Neurocomputing*, 580 – 589.
- [26] Zhang, J., Zheng, Y., Sun, J., & Qi, D. (2019). Flow prediction in spatio-temporal networks based on multitask deep learning. *IEEE Transactions on Knowledge and Data Engineering*, 32(3), 468-478.
- [27] Chen, C., Li, K., Teo, S. G., Zou, X., Li, K., & Zeng, Z. (2020). Citywide traffic flow prediction based on multiple gated spatio-temporal convolutional neural networks. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 14(4), 1-23.
- [28] Liu, Y., Zheng, H., Feng, X., & Chen, Z. (2017, October). Short-term traffic flow prediction with Conv-LSTM. In *2017 9th International Conference on Wireless Communications and Signal Processing (WCSP)* (pp. 1-6). IEEE.
- [29] Zhang J, Zheng Y, Sun J, et al. Flow prediction in spatio-temporal networks based on multitask deep learning[J]. *IEEE Transactions on Knowledge and Data Engineering*, 2019, 32(3): 468-478.
- [30] Zhang, J., Zheng, Y., & Qi, D. (2017, February). Deep spatio-temporal residual networks for citywide crowd flows prediction. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 31, No. 1).
- [31] Guo G, Zhang T. A residual spatio-temporal architecture for travel demand forecasting[J]. *Transportation Research Part C: Emerging Technologies*, 2020, 115: 102639.
- [32] Defferrard, M., Bresson, X., & Vandergheynst, P. (2016). Convolutional neural networks on graphs with fast localized spectral filtering. *Advances in neural information processing systems*, 29.
- [33] Hamilton, W., Ying, Z., & Leskovec, J. (2017). Inductive representation learning on large graphs. *Advances in neural information processing systems*, 30.

- [34] Yu, B., Yin, H., & Zhu, Z. (2017). Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. arXiv preprint arXiv:1709.04875.
- [35] Chai, D., Wang, L., & Yang, Q. (2018, November). Bike flow prediction with multi-graph convolutional networks. In Proceedings of the 26th ACM SIGSPATIAL international conference on advances in geographic information systems (pp. 397-400).
- [36] Cao, X., Zhong, Y., Zhou, Y., Wang, J., Zhu, C., & Zhang, W. (2017). Interactive temporal recurrent convolution network for traffic prediction in data centers. IEEE Access, 6, 5276-5289.
- [37] Tormene, P., Giorgino, T., Quaglini, S., & Stefanelli, M. (2009). Matching incomplete time series with dynamic time warping: an algorithm and an application to post-stroke rehabilitation. Artificial intelligence in medicine, 45(1), 11-34.
- [38] Deng, A., & Hooi, B. (2021, May). Graph neural network-based anomaly detection in multivariate time series. In Proceedings of the AAAI conference on artificial intelligence (Vol. 35, No. 5, pp. 4027-4035).
- [39] Szegedy C, Ioffe S, Vanhoucke V, et al. Inception-v4, inception-resnet and the impact of residual connections on learning[C]//Proceedings of the AAAI conference on artificial intelligence. 2017, 31(1).
- [40] Shi, X., Qi, H., Shen, Y., Wu, G., & Yin, B. (2021). A Spatial – Temporal Attention Approach for Traffic Prediction. IEEE Transactions on Intelligent Transportation Systems, 4909 – 4918.
- [41] Tang, J., & Zeng, J. (2022). Spatiotemporal gated graph attention network for urban traffic flow prediction based on license plate recognition data. Computer-Aided Civil and Infrastructure Engineering, 37(1), 3 – 23.
- [42] Posner, M. I. (1994). Attention: the mechanisms of consciousness. Proceedings of the National Academy of Sciences, 91(16), 7398 – 7403.
- [43] Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., & Adam, H. (2018). Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. In Computer Vision – ECCV 2018, Lecture Notes in Computer Science (pp. 833 – 851).
- [44] Yuan, Y., Xiong, Z., & Wang, Q. (2019). VSSA-NET: Vertical Spatial Sequence Attention Network for Traffic Sign Detection. IEEE Transactions on Image Processing, 3423 – 3434.
- [45] Juddoo, S. (2022). Investigating Data Repair steps for EHR Big Data. 2022 3rd International Conference on Next Generation Computing Applications (NextComp). Presented at the 2022 3rd International Conference on Next Generation Computing Applications (NextComp), Flic-en-Flac, Mauritius.
- [46] Du, Y., Liu, Y., Peng, Z., & Jin, X. (2022). Gated attention fusion network for multimodal sentiment classification. Knowledge-Based Systems, 240, 108107.
- [47] Huang, Z., Wang, X., Huang, L., Huang, C., Wei, Y., & Liu, W. (2019). CCNet: Criss-Cross Attention for Semantic Segmentation. 2019 IEEE/CVF International Conference on Computer Vision (ICCV). Presented at the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea (South).
- [48] Srivastava, N., Hinton, Geoffrey E., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. Journal of Machine Learning Research, Journal of Machine Learning Research.
- [49] Zhang, Y., Wang, S., Chen, B., Cao, J., & Huang, Z. (2021). TrafficGAN: Network-Scale Deep Traffic Prediction With Generative Adversarial Nets. IEEE Transactions on Intelligent Transportation Systems, 219 – 230.
- [50] Yao, H., Tang, X., Wei, H., Zheng, G., & Li, Z. (2019). Revisiting Spatial-Temporal Similarity: A Deep Learning Framework for Traffic Prediction. Proceedings of the AAAI Conference on Artificial Intelligence, 5668 – 5675.

- [51] Scarselli, F., Gori, M., Ah Chung Tsoi, Hagenbuchner, M., & Monfardini, G. (2009). The Graph Neural Network Model. *IEEE Transactions on Neural Networks*, 61 – 80.
- [52] Kipf, T., & Welling, M. (2016). Semi-Supervised Classification with Graph Convolutional Networks. *arXiv: Learning*, arXiv: Learning.