University of Macau

Faculty of Science and Technology



Your Project TitleTraffic Flow Forecasting with

Deep Graph Convolution Network

by

Your NameYour NameIp Pak Lon, Student No: DB625471Dxxxxxx

Final Project Report submitted in partial fulfillment  
of the requirements of the Degree of   
Bachelor of Science in Computer Science

Project Supervisor

Prof. Ryan L.H.U

04 June 2020

# 

DECLARATION

I sincerely declare that:

1. I and my teammates are the sole authors of this report,
2. All the information contained in this report is certain and correct to the best of my knowledge,
3. I declare that the thesis here submitted is original except for the source materials explicitly acknowledged and that this thesis or parts of this thesis have not been previously submitted for the same degree or for a different degree, and
4. I also acknowledge that I am aware of the Rules on Handling Student Academic Dishonesty and the Regulations of the Student Discipline of the University of Macau.

Signature : \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Name : Ip Pak Lon

Student No. : DB625471

Date : 22 May 2020

ACKNOWLEDGEMENTS

I would like to express my utmost gratitude to UM for providing the opportunity to carry out a project as partial fulfillment of the requirement for the degree of Bachelor of Science.

Throughout this project, I was very fortunate to receive the guidance and encouragement from Prof.Ryan Leong Hou U, my research supervisor, he provided valuable and constructive suggestion during the project, His willingness to give his time so generously to discussion with me has been very much appreciated.

**ABSTRACT**

Macau is the most densely populated city with the highest motor vehicle density in the world, the population of humans grows beyond the carrying capacity of their environment. Macau government is increasingly aware of the concept of ‘smart city’ and is additively developing a strategy to solve the serious traffic problem and to achieve the goals of becoming ‘smart’. With the development of the technology of Artificial Intelligence and Big data, big Spatio-temporal data-driven smart traffic management becomes possible, many aspects of traffic problems can be further optimized.

This project focus on processing the Spatio-temporal data for forecasting of traffic-flow with Deep Learning method. Traditional statistical time series prediction methods such as HA, ARIMA are not suitable for long term prediction tasks, Deep learning approaches for times series prediction such as RNN and LSTM can capture the long-term dependencies and solve the gradient vanishing problem during long propagation, however, it neglects the dependencies of spatial and temporal which is a significant feature of traffic flow prediction. Urban traffic data has two important features, one is the periodicity on time and the connective on spatial. Since the traffic network is on a non-euclidean domain, traditional convolution can not apply to graph data since the kernel does not fulfill the requirement of translation invariance. Therefore Graph Convolution Network (GCN) is introduced to perform the convolution on the graph. A Spatio-temporal graph convolution(ST-GCN) module is proposed to dynamically model the urban road traffic-flow. It shows a promising result, however, most of the implementation builds a very shallow model inspired by the residual network(he kai ming), the convolution network should be built deeper and it will get a better result, however, due to the gradient vanishing and over-smoothing on graph message passing problem, most SOTA models are no deeper than 4 layers. This work aims to figure out the solution and find the most optimized model with different structures.

Table of Contents

[CHAPTER 1. INTRODUCTION 9](#_Toc41858987)

[1.1 Overview 9](#_Toc41858988)

[1.2 Problem Statement 9](#_Toc41858989)

[CHAPTER 2. Methodology 11](#_Toc41858990)

[2.1 GCN 11](#_Toc41858991)

[2.1.1 Graph Laplacian 11](#_Toc41858992)

[2.1.2 Fourier Transform on Graph 11](#_Toc41858993)

[2.1.3 Varients of Graph Convolution 11](#_Toc41858994)

[2.2 Temporal convolution 12](#_Toc41858995)

[2.3 Residual connection 12](#_Toc41858996)

[2.3.1 Identity mapping 13](#_Toc41858997)

[CHAPTER 3. Propose model 14](#_Toc41858998)

[3.1 Traffic speed forecasting 14](#_Toc41858999)

[3.2 GCN block 14](#_Toc41859000)

[3.3 TCN block 14](#_Toc41859001)

[3.4 STGCN block 15](#_Toc41859002)

[3.5 The STGCN network 16](#_Toc41859003)

[3.6 The characteristic of the model 16](#_Toc41859004)

[3.6.1 General model 16](#_Toc41859005)

[3.6.2 Fully build on convolution 16](#_Toc41859006)

[CHAPTER 4. Experiment 17](#_Toc41859007)

[4.1 Dataset 17](#_Toc41859008)

[4.2 Experiment Settings 17](#_Toc41859009)

[4.3 Result 18](#_Toc41859010)

[4.3.1 Effect of residual connections 18](#_Toc41859011)

[4.3.2 Performance Comparision 19](#_Toc41859012)

[CHAPTER 5. Related Work 20](#_Toc41859013)

[CHAPTER 6. Conclusion and future works 21](#_Toc41859014)

[CHAPTER 7. References 22](#_Toc41859015)

LIST OF FIGURES

[Figure 1: Causal Deligatied CNN. 11](#_Toc375841968)

[Figure 2: Residual connection 12](#_Toc375841969)

[Figure 3: Spatio-temporal graph 13](#_Toc375841970)

[Figure 4: Bottleneck structure 14](#_Toc375841971)

[Figure 5: Spatio-temporal block 14](#_Toc375841972)

[Figure 6: Structure of STGCN network 15](#_Toc375841972)

[Figure 7: The training loss and validation loss of deep STGCN 17](#_Toc375841972)

[Figure 8: The training loss and validation loss of deep Residual STGCN 17](#_Toc375841972)

LIST OF TABLES

[Table 1: The evaluate result of different models 12](#_Toc375841973)

# INTRODUCTION

## Overview

The extensive use of intelligent portable devices brings convenience to traffic data, combining with the smart sensor on the road, a large range of traffic data can be efficiently collected for further analysis in real-time. time-division sampling of traffic-flow data can help to understand the congestion level of the urban road map and help to adjust the traffic arrangement system, such as traffic-light scheduling and public transportation arrangement.

In traffic study, speed, volume, and car density are usually used to indicate the status of traffic flow. Based on the length of prediction, traffic forecasting can be classified to short-term forecasting(less than 30 mins) and long-term forecasting(over 30 mins), traditional statistical methods perform well in short-term forecasting but it is not able to capture the long-term dependencies such as workday and holiday periodicity. Deep learning approaches have gained great success in various traffic tasks, The recurrent neural network performs well on Forecasting based on historical data of individual road. With the gate mechanism[1], long-term features can be learned well and successfully capture the periodicity and regularity of time-series data. However, the transmissibility on the graph of traffic flow is not to be neglected. For example, when a road is congested, the congestion will be diffuse to its up-flow and down-flow road. The congestion will be further defused to its neighbored road respectively. It likes heat exchange progress, the congestion will be diffused to the whole road network at the end unless the congestion is digested by the spare roads. it is a tough work to learn the propagation of road network, some studies separate the road network as different segmentation grid, then use CNN on the grid map to learn the propagation, regard the road network as an image so that the technique of CNN on the image can be used to learn the propagation method however it omits the graph structure of road network.

GCN[2] has been proved to be a successful module to learn the probation on the graph structure, it is suitable to apply on the road network, but GCN requires the adjacency matrix to be symmetric, since the road network is a directed graph, so it can be decomposed by the Laplacian matrix, in this work, the direction is omitted, the road network is adjusted to be a symmetric undirected graph. we construct a Spatio-temporal convolution network with deeper layers base on the ST-GCN module[3].

## Problem Statement

Traffic speed forecast is a typical time-series prediction problem, it forecast the future speed based on historical data and the dependencies of the road network. The traffic road network can be defined as a graph structure with connecting roads, the roads as the edge, and the connection points as the nodes in the graph. The separation of the road is base on the sensor of the roads. The speed data is collected from these sensors at a time. For the data cleaning and enhance the stability, the data is fused to 5 mins data with the average value. The historical speed data of every node construct the Spatio-temporal graph data.

The historical Spatio-temporal graph is further used to forecast the future traffic-flow, the task is to forecast the traffic flow on the whole road network instead of predicting one specific road.

Spectral-domain GCN especially the ChebyNet[4] and it is first approximation[2] is widely used on non-euclidean data processing, such as graph clustering and traffic-flow prediction. A key reason why CNN gains great succeed in different learning tasks is that CNN can be built as a very deep model(hundreds layer), with a deeper layer, the model can learn more detail and become more complex to capture the feature. The proposal of Residual network structure [5] makes it possible to train a very deep CNN. However, there is no mechanism for training a deep GCN, with deeper GCN, the value of each node on the graph will tend to be equal due to the gradient vanishing problem[6]. This project aims to compare different structures of making GCN deeper on the traffic flow forecast through experiments.

# Methodology

## GCN

### Graph Laplacian

Inspired by Fourier transform, if a set of orthogonal bases can represent the graph, these orthogonal bases can be treated as the spectral of the graph. Fortunately, the graph laplacian makes this idea possible.

Graph Laplacian is widely used in graph theory, as a representation of a graph it is defined as , D is the degree matrix which is a diagonal matrix with the value of the degree of the vertex, A is the adjacency matrix. GCN uses The symmetric normalized form of Laplacian matrix . The Laplacian matrix of an undirected graph has 3 properties: (1) it is a positive semi-definite matrix so that it is eigenvalue is non-negative; (2) the amount of zero eigenvalues is the number of connectives area of the graph; (3) the value of the smallest non-zero eigenvalues is the algebraic connectivity of the graph.

Since Graph Laplacian is a positive semi-definite matrix, it can be used to do eigendecomposition, the Eigenvectors span linearly independent and orthogonal bases and can be treated as the spectral of the graph.

### Fourier Transform on Graph

The matrix form of Fourier transform on a graph is defined as , U is the matrix of eigenvectors. The inverse Fourier transform is defined as According to the convolution theory, Convolution in the time domain equals multiplication in the frequency domain. The convolution of the graph on the spectral domain is much easier the convolution on Fourier transform is defined as . Graph convolution can be defined as .

### Varients of Graph Convolution

The most primitive version of graph convolution is defined as . The is a function of the eigenvalue of the graph which is the kernel of the convolution, the kernel size is equaled to the nodes of the graph. The calculation of the eigenvector matrix multiplication and the eigendecomposition of the laplacian matrix is large, so it is not an effective method.

M.Defferard uses the Chebyshev polynomial of the eigenvector matrix as the kernel filter to propose Chebnet [7]Chebyshev polynomial is recursively defined as The formula of convolution is given as , is the matrix of eigenvalue after scale to [-1,1], it is to fulfill the requirements of Chebyshev polynomial Kth truncate. This expression is K-localized, and it is local connectivity, Since it is the kth stage polynomial of Laplacian operator, the propagation depends on the k step neighbor of the central nodes.

T.N.Kipf propose a simpler and effective layers propagation method 1st ChebNet [2] it let the k=1 and , the ChebNet convolution can be approximated as , 1st ChebNet also suppose . Then the formula can be further simplified as , however, is in the range [0,2] if it is used in a deep neural network, it will meet the gradient explosion problem. To solve this problem Kipf propose a renormalization trick by adding self-connection on a graph: Let then , combine with an activation function the fast graph convolution formula is given by )

## Temporal convolution

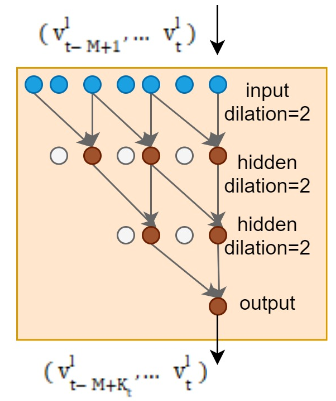
Although the Deep learning model base on RNN can be widely used on time series analysis, applying RNN on traffic forecast is a time-consuming process, it has a complex gate control mechanism, it can not timely react to dynamic changes. We propose a stacked deligated convolution model with gate control. It has 3 pros: (1) the low-level CNN captures the dependencies of the nearer time step, high-level CNN captures the dependencies of the further time step, base on this layer-rise structure, it’s more likely to capture long-term dependencies. (2) stacked CNN only takes n/k process comparing with RNN of k processes, the efficiency is much enhanced (3) Since CNN does not rely on the historical information of the series, it can be parallel calculated. The training speed is much faster, can handle larger data and fulfill the real-time processing requirement.

Figure 1: Causal Deligatied CNN

The temporal convolution uses causal deligated convolution purpose in Wavenet(Deep mind et.al 2016) it stacks deligated temporal CNN layer to build the block. It increases the reception field and reduces the calculation The filter is flexible to change by tuning the parameter of dilation and the level of layers.

## Residual connection

As the neural network goes deeper the relativity of the back propagation gradient will decrease, and approximate to white noise at the end. If the gradient is approximate to white noise, training a network using that gradient is without distinction to random training. Although using a normalized trick such as the Batch normalization layer[8] can stabilize the gradient to normal range, prevent the occurrence of gradient descending and gradient explosion, however, the relativity of gradient is still attenuating as the layers increase.

### Identity mapping

The ResNet model [2]presents a solution by added identity mapping layers and copied the learned shallower model to construct a deeper model. As the model goes deeper, the model will be degraded, adding identity mapping to map the shallow layer to deep layer can reduce the influence. The propose of residual learning is to let the inter-structure of the deep model have the ability of identity mapping, it can prevent the degradation of the model during the stacked-layer to a deeper model.

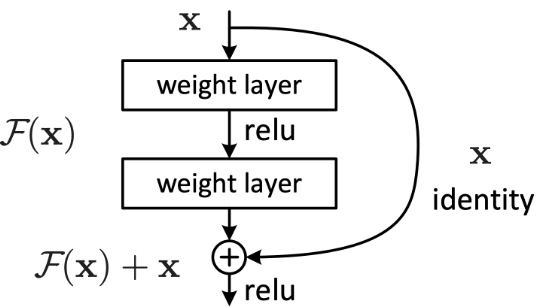


Figure 2: Residual connection

# Propose model

In this section, we will define the traffic speed forecasting task and give a detailed description of the STGCN network structure.

## Traffic speed forecasting

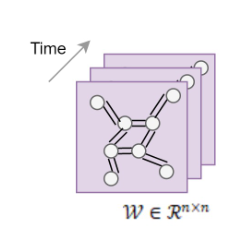
We aim to predict future traffic speed on every sensor on the traffic network, the input is a list of Spatio-temporal graph defined as the figure3. Each Spatio-temporal graph contains the sensors as nodes and road as vedges. Each node contains a feature vector to describe the network status, the feature vector can contain average traffic speed, traffic congestion, car flow, etc. in our experiment we only use the average speed in 5mins as the feature. By learning the pattern of the historical Spatio-temporal graph the model aims to predict the future historical Spatio-temporal graph.

Figure 3: Spatio-temporal graph

## GCN block

The Graph convolution block use the ChebyNet module with kernel k as the graph convolution block, the change of conception field of graph convolution is implemented by adjusting the kernel size, the convolution is written as:

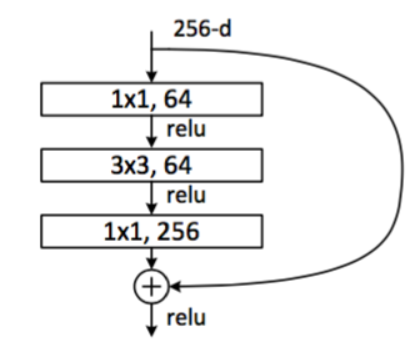
is the matrix of eigenvalue after scale to [-1,1]. We first produced the scaled laplacian base on the adjacency matrix, then use Eigendecompositon to get the eigenvalue and eigenvector of spectral representation of the graph. The GCN is a linear computation, the calculation cost is in O(K|E|). In this work, we explored the effect of deep GCN block with and without residual connection. Try to find optimize level layers of GCN on a small graph.

## TCN block

The temporal convolution block use the causal deligated convolution on time series, instead of using RNN which is very popular in time series analysis, using convolution can enhance the reaction speed and reduce the training difficulty of the model, besides, using causal and deligated structure can significantly reduce the computation cost and more likely to learn long term dependencies. The temporal graph contains a 4-layer causal deligated convolution follow by a Gated Linear Unit as nonlinearity. The output of the convolution is separated into two parts [P,Q] with the same channels, then utilizes [P,Q] as the input of the GLU to perform non-linearity. The formula is given as below

Where represent the kernel, x is the time series input, ,,, are model parameters g is an activation function, we use ReLU in this model, is the sigmoid activation model.

## STGCN block

To learn the spatial and temporal dependencies, The STGCN combine TCN and GCN to construct an ST module, The STGCN block is built as a sandwich likes structure which is inspired by the bottleneck structure of ResNet, the top and the bottom set to be TCN block and stack n layers of GCN block inside. The block can be stacked or extended to different scales and complexity cases.

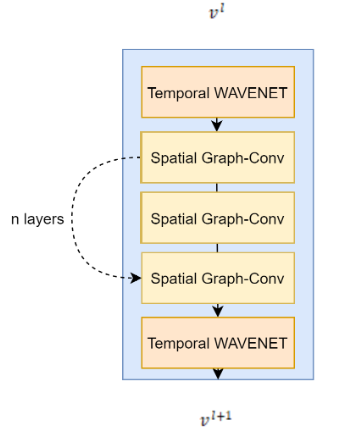
Figure 4: Bottleneck structure

Figure 5: Spatio-temporal block

## The STGCN network

The STGCN network contains two STGCN blocks followed by a fully connected output layer, the input of the model contains two parts, one is the M time step Spatio-Temporal graph which contains n feature vectors , is the input feature size which set to be 1 since the average speed is the only considered feature in our work. The other part is the adjacency matrix which is utilized to get the graph laplacian and do Eigendecomposition.

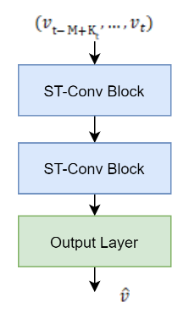


Figure 6: Structure of STGCN network

## The characteristic of the model

### General model

STGCN is a general model for structure time series, it is appropriate for extracting space connection information and capture time-series feature, besides of traffic-flow forecast task, the module can be applied to any structure Spatio-temporal data.

### Fully build on convolution

The TCN and GCN are all built on convolution, Convolution has the advantage of fast training and simpler architecture, it has fewer parameter than applying gate control in RNN, it is more likely to convergence, in the experiment the model convergence to a stable result after less than 50 epoch, CNN can also achieve parallelization and fast processing, it can maximize the power of GPU and handle larger network with fewer reaction time, it brings the possibility to real-time processing in industry production.

# Experiment

In this work, we compare the effect of stacking different depth of GCN and reconstruct the network using ResNet by adding residual connection and demonstrate the significant improvement of deep ResNet GCN, the baseline of the experiment is the STGCN model with 2 Spatio-temporal blocks containing 2 TCN and 1 GCN inside. We compared the model with classical statistical methods such as HA, ARIMA, and some simple Deep Learning models such as FNN and LSTM. The result shows the superiority result of our purpose model.

## Dataset

We verify our model on real-world traffic datasets PEMSD7 which is collected by the Caltrans Performance Measurement System (PeMS) in real-time by over 39, 000 sensor stations. The dataset contains including a weighted adjacent matrix and historical speed data for 228 sensors. The dataset is fused to 5mins data base on the average speed of 30s data samples. The experiment picks the district 7 of Califonia containing 228 sensors as the data. The dataset only contains the weekday of May and June of 2012, the dataset is separated in 3 parts the first 34 days data as the training dataset, 5 days data for validation, and 5 days data for evaluating.

## Experiment Settings

The experiment is compiled on a personal computer with windows 10(CPU: intel® core(TM) i7-9700K @3.60 GHz, GPU: two NVIDIA GeForce RTX 2080) all the test using 12-time slots data to predict 3-time slots data in the future, in the other word, it predicts the future (15,30,45 mins) data base on 60 mins historical data. The depth is set to be 8,16,32,56 layers of deep GCN respectively.

To evaluate the performance of the models, we adopt 3 indicators which are widely used on predicting problem, Mean Absolute Errors (MAE), Mean Absolute Percentage Errors(MAPE), and Root Mean Squared Error(RMSE). We compared the different depth of GCN and ResGCN with the following Statistical model and simple deep learning baseline: 1) Historical Average 2) Auto-Regressive Integrated Moving Average 3) simple Fully connected Feed Forward Neural Network 4) fully connected LSTM(Sutskever et al., 2014).

## Result

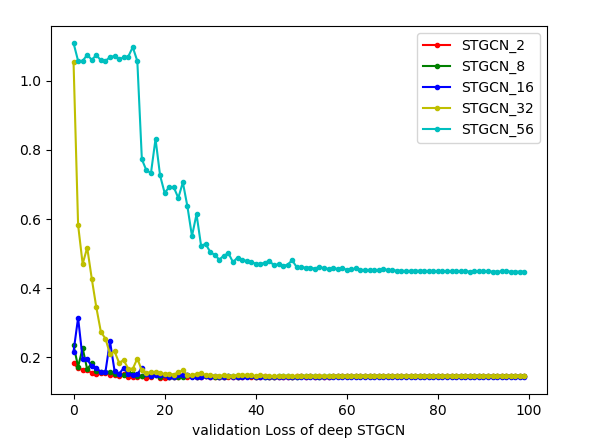
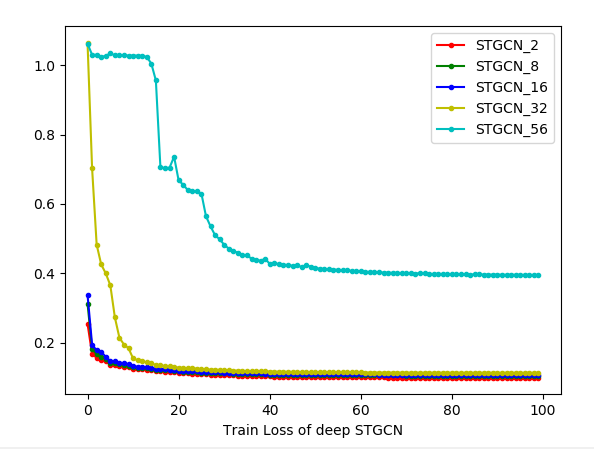
We investigate the performance of different depth of GCN, the result has shown that the training loss and validation loss is increased as the layer increase, The model still can convergence to a good result when the depth under 32, when applying the 56 layers GCN the result significantly deteriorate. This phenomenon is mainly caused by the gradient vanishing during the propagation of gradient in such a deep model. Therefore base on the research of ResNet adding residual connection can intuitively solve this problem.

Figure 7: The training loss and validation loss of deep STGCN

### Effect of residual connections

The residual connection has been proved as an essential role in training deeper network, after constructing the GCN with residual connection, the result is more stable comparing with platin deep GCN. It solved the gradient vanishing problem on the deep network, therefore the stacked of deep GCN become an available strategy.

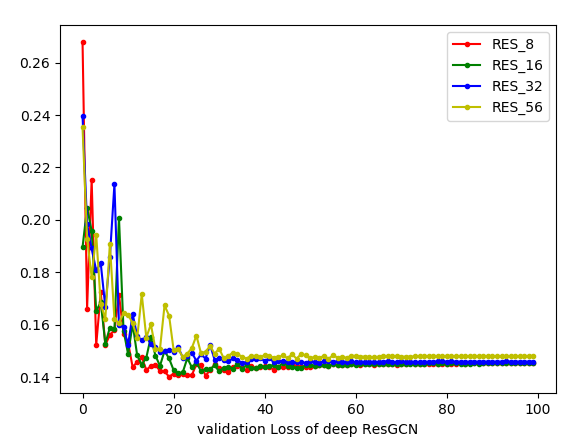
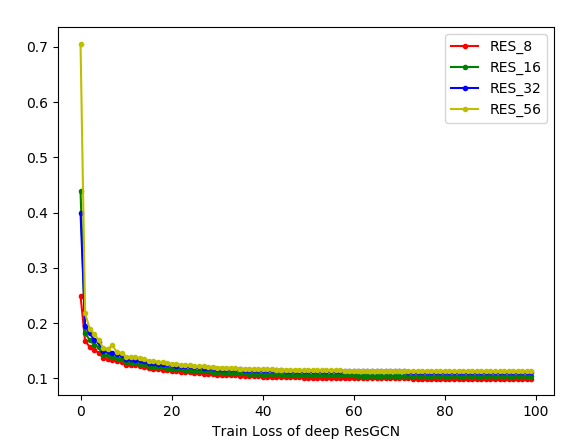
**

Figure 8: The training loss and validation loss of deep Residual STGCN

### Performance Comparision

The experiments of exploring the depth shown that the STGCN with 2 GCN layers and RESGCN with 8 layers get the best result on the datasets. Although using the ResNet structure solved the gradient descending problem, the performance does not monotone increasing with stacked CNN layers, the model gets the best result with 16 layers then the performance gets worse with the layer increased, there are 3 main problems: 1) The dataset only contains 228 nodes on the graph, the propagation does not need to go very deep to traverse on the whole graph, the kernel of Chebyshev polynomial is set to be 3 which means the 3 nearest neighbors are considered in one convolution, with 8 layers the message passing of the nodes gets the best result in the experiment. 2) The dataset is very dense since the road segment we pick is in center of the city, it does not need a very long message passing to traverse on the graph; 3) Applying GCN is similar to the heat-exchange process, applying deep GCN on a dense graph will cause an over-smooth problem, that is the value of each nodes convergence to similar value at the end.

|  |  |  |  |
| --- | --- | --- | --- |
| Model | PEMSD7 | | |
| MAE(15/30/45mins) | MAPE(15/30/45mins) | RMSE(15/30/45mins) |
| HA | 4.01 | 10.61 | 7.20 |
| ARIMA | 5.55/5.86/6.27 | 12.92/13.94/15.2 | 9.00/9.13/9.38 |
| FCNN | 2.74/4.02/5.04 | 6.38/9.72/12.38 | 4.75/6.98/8.58 |
| FC-LSTM | 3.57/3.94/4.16 | 8.60/9.55/10.10 | 6.20//7.03/7.51 |
| STGCN-2 | 2.244/2.974/3.685 | 5.264/7.071/8.892 | 3.987/5.577/7.011 |
| RESGCN-8 | 2.243/2.987/3.3615 | 5.238/7.097/8.322 | 3.960/5.585/6.137 |

We pick STGCN-2 and RESGCN-8 to compare with the baseline, the result is shown in table 1:

Table 1: The evaluate result of different models

As the table is shown, the traditional statistical method performs worst in the experiment, the ARIMA method gives a stable result on the short-term and long-term series prediction task, FCNN performs well in short term prediction while LSTM gives a better result on long term prediction, for short term prediction, the performance of the deep model and shallow model is similar, the superiority of deep GCN is reflected in long term prediction, the traffic information of neighbor node plays an important role in long term prediction, our model with residual connection gives the best result in the experiment it proves the fact that a deeper GCN model can better help to extract the spatial feature.

# Related Work

The graph convolution network is widely used in many kinds of research of spatial relationship modeling, especially the 1st approximation introduced by T.N.Kipf, base on the GCN, modify the TCN model can fit the model into different scales or types of cases. Recurrent neural networks (RNNs) or its variants are often combined with the encoded spatial relationship to model temporal dependencies (Yu, Yin, and Zhu 2018; Gehring et al. 2017).

Graph convolution on the spatial domain is also a popular direction to capture the spatial feature of the graph. (Li et al. 2018) present the diffusion convolution recurrent neural network (DCRNN) which combines diffusion convolution and recurrent neural networks for modeling spatial and temporal dependencies. Inspiring by attention mechanism from NLP, (Park et al. 2019) present the Spatio-Temporal Graph Attention Network (STGAT), instead of learning the propagation on the whole graph, node attention is utilized to capture the spatial correlation among roads and get the state of the art to result in 2019 with significant improvement.

# Conclusion and future works

In this work, we utilize the causal deligated convolution to capture long-term dependencies on the time domain, There are many studies on time series analysis, learning long-term dependencies is no longer a difficult task, however, some kinds of time-series data always perform randomnesses such as the forecast of stock(political changes, market sentiment or sudden news) and traffic-flow prediction(car accident, the hold of important activities). The use of periodicity on time series can not fit well in this situation. Traffic forecasting should consider some hyper hidden information such as the weather, hold of important activity, real-time car accident on the road, etc.

Our work introduced an available method to build a deep model by stacking GCN layers and add the residual connection. Deep GCN model has a wider conception field, the model can learn further propagation pattern, however, further propagation along with the road network for traffic network did not monotone increase the performance on traffic forecast work in our experiment. There are serval reasons to conclude, the traffic flow does not propagate to far nodes, the road itself can defuse the traffic flow in a few steps. Since the eigendecomposition requires the graph to be symmetric, however, the up-flow and down-flow of the road network should not be treated as equal, besides there are many one-way roads on the traffic network which mean the adjacency matrix itself should not be considered as a symmetric matrix. The propagate pattern of traffic flow is more similar to diffusion on a tree structure compared to diffusion on all neighbors as the behavior of interchange of heat.

The time series analysis is a sequence to sequence model, which is widely used on nature language processing area, some seq2seq model from NLP is also suitable to apply on traffic-flow forecasting task, for instance, [9] introduce a spatial-temporal graph attention network for traffic forecasting, the attention mechanism is first to propose from the NLP area, or [10] utilize the transformer structure to compose a traffic transformer network for traffic-flow forecasting. The transformer structure is another popular model in NLP in addition to auto-encoder.

The traffic graph can be analyzed as a heterogeneous model since the road has a different level(sideway, lane, main road, etc) also the propagation on the graph can be divided into a hierarchical model(from node level to area level). With adding meta-path on the sentimental level, the model can perform a hierarchical attention mechanism to learn the embedding representation on the layer-rise traffic-flow propagation pattern.

.

# References

1. Xingjian Shi, Zhourong Chen, Hao Wang, Dit-Yan Yeung, Wai-Kin Wong, and Wang-chun Woo. "Convolutional lstm network: A machine learning approach for precipitation nowcasting". In NIPS, pages 802–810, 2015
2. Thomas N Kipf and Max Welling. "Semi-supervised classification with graph convolutional networks". arXiv preprint arXiv:1609.02907, 2016.
3. Yu, B.; Yin, H.; and Zhu, Z. 2018. "Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting". In Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)
4. David K. Hammond, Pierre Vandergheynst, and Remi Gribonval. "Wavelets on graphs via spectral graph theory". Applied and Computational Harmonic Analysis, 30(2):129–150, 2011.
5. K. He, X. Zhang, S. Ren, and J. Sun. "Deep residual learning for image recognition". In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 770–778, 2016. 1, 2, 4, 6
6. Q. Li, Z. Han, and X.-M. Wu. "Deeper insights into graph convolutional networks for semi-supervised learning". In Thirty-Second AAAI Conference on Artificial Intelligence, 2018. 1, 2, 3
7. Michael Defferrard, Xavier Bresson, and Pierre Vandergheynst. "Convolutional neural networks on graphs with fast localized spectral filtering". In Advances in neural information processing systems (NIPS), 2016.
8. Sergey Ioffe, Christian Szegedy “Batch Normalization: Accelerating Deep Network Training b y Reducing Internal Covariate Shift”, 2015
9. ChenHan Zhang , James J. Q. Yu , (Member, IEEE), AND YI LIU “Spatial-Temporal Graph Attention Networks: A Deep Learning Approach for Traffic Forecasting”, 2019
10. Mingxing Xu, Wenrui Dai, Chunmiao Liu, Xing Gao, Weiyao Lin, Guo-Jun Qi, Hongkai Xiong “Spatial-Temporal Transformer Networks for Traffic Flow Forecasting”,2020