COMP4434 Course Project Report

YU Jing 16098537d
HAO Shiqi 16098696d
JIANG Yuxin 16096336d
WANG Jiashuo 16096527d
WANG Bokang 16097234d
CHENG Yiran 16098521d
Group 3

1 Background

The flow of crowds in a city is an important indicator of traffic management and public safety. Nowadays, traffic congestion during the rush hour is a serious problem in Chinese metropolises. Massive crowds of people or cars streamed into a small region may result in catastrophic stampede or traffic accidents. Therefore, predicting the crowd flow in the upcoming time period can effectively ease traffic jams, greatly reduce traffic accidents and even prevent disasters, which is helpful to maintain traffic order and safety.
In this project, we design and test a model to predict the flows in next time slot. The model uses the last inflows and outflows of a grid as well as meteorological data.

2 Problem Definition

2.1 Problem Specification

Design and test a model which applies last inflows/outflows of a grid to predict the flows in next timeslot. Using meteorological data may improve the prediction accuracy.

2.2 Notation Definition

Region:

$$(i, j)$$

A grid cell lies at the i-th row and j-th column in the map, represented as (i, j).
Inflow:

$$x_t^{in,(i,j)}$$

Inflow of the crowds at the time t for the region (i, j).
Outflow:

$$x_t^{out,(i,j)}$$

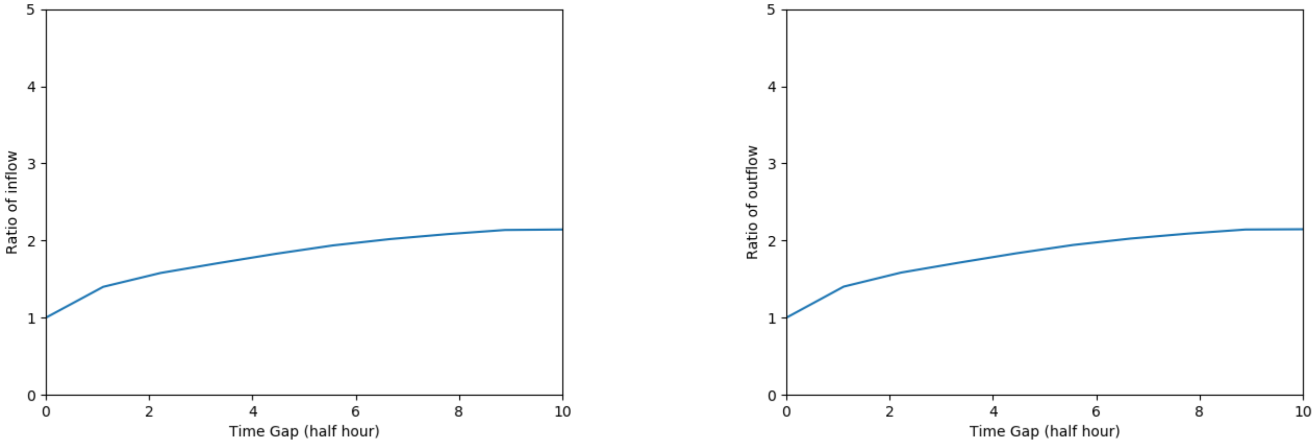
Outflow of the crowds at the time t for the region (i, j).

3 Model Design and Considerations

3.1 Raw Data Analysis

Before choosing the input data and dimension, we analyzed raw data at the beginning.
Spatial dependency: It is an empirical finding that the flows of a region will be affected by that of the nearby regions. For example, the inflow of a region, (i, j), is affected by the outflows of its nearby regions, such as (i+1, j), (i-1, j), (i, j+1), etc. Similarly, its outflow may affect inflows of other nearby regions, like (i, j-1). In addition, the inflow of a region will also affect its own outflow.
Temporal dependencies: This feature is mainly reflected in two aspects, one is that adjacent time slot affects each other called closeness in this report. The other is that inflow and outflow have periodic trends called period.

- *Closeness*: The flows (inflows and outflows) of all regions are influenced by the adjacent time slots. It is showed in the Fig.1 that when the time gap is small before 2-hour, the average ratio which is the value between arbitrary two inows that have the same time gap is small enough to display the correlation. For example, the inflows and outflows in 20:00 pm will be influenced by them in recent time such as 19:30 pm and 19:00 pm.Moreover, the smaller time gap implies a stronger influence.

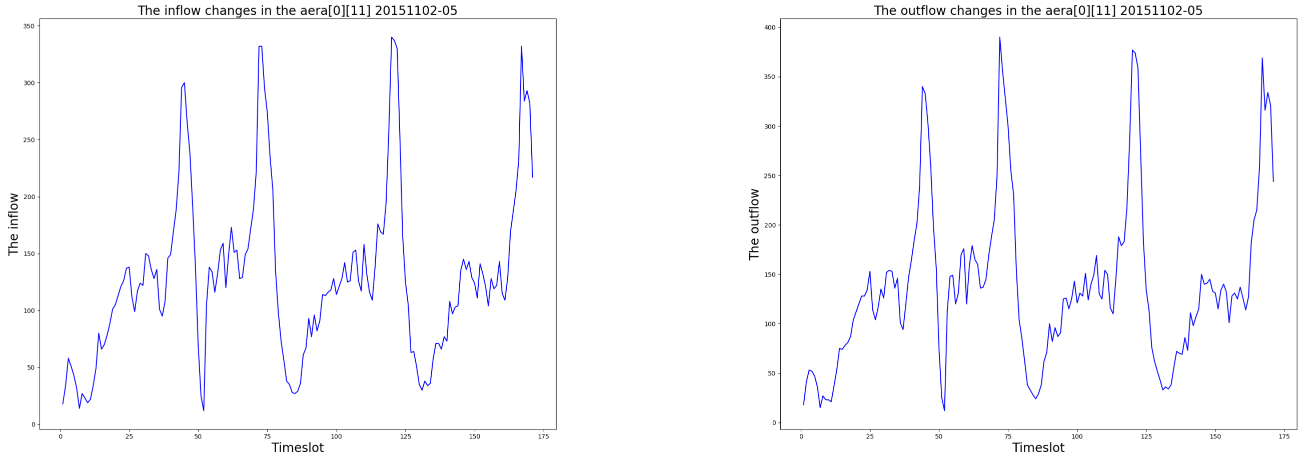


(a) inflow

(b) outflow

Figure 1: The Closeness Feature

- *Period*: The curves in the Fig.2 Traffic conditions may be similar on consecutive weekdays (every 24 hours).



(a) inflow

(b) outflow

Figure 2: The Period Feature

External influence: The external influence such as the weather condition, temperature and wind speed may also hold an impact on the crowd flows. For instance, when the temperature is low, there are fewer people hang out and the peak point will push back.

3.2 Input Data

Based the features of raw data which have been mentioned above, the total dimensions of input data are $10 * 32 * 32$:

Data Type	Time Scale	Area Range	Dimensions
The inflow	Three time slots before the predicted	32*32	3*32*32
The outflow	Three time slots before the predicted	32*32	3*32*32
Temperature	Predict time slot	32*32	1*32*32
Wind Speed	Predict time slot	32*32	1*32*32
The inflow	The same time slot of a day before predict day	32*32	1*32*32
The outflow	The same time slot of a day before predict day	32*32	1*32*32

There is an example which explained the input data chosen:
If the inflow and outflow which need to predict is at time 2015-11-02-10 and then the input data will bet inflows and outflows of 2015-11-02-09, 2015-11-02-08 and 2015-11-02-07, the temperature of 2015-11-02-10, the wind speed of 2015-11-02-10 and the inflow and outflow of 2015-11-01-10.

3.3 Methodology

To decide what model to use, characteristics of input and output should be analyzed. As mentioned, the crowd flows are spatial dependencies and temporal dependencies. The inputs *timeslots* provide the temporal features. Therefore, a method that can capture spatial structures should be adopted in this model. Thus, we use CNN to build the model.

A **convolutional neural network (CNN)** with convolutional layer and activating layer (ReLU) is used. **CNN** has a powerful ability to a).hierarchically capture the spatial structural information, and b).deal with inputs with high dimensionality. Figure 3 presents the structure of the final model in this project.

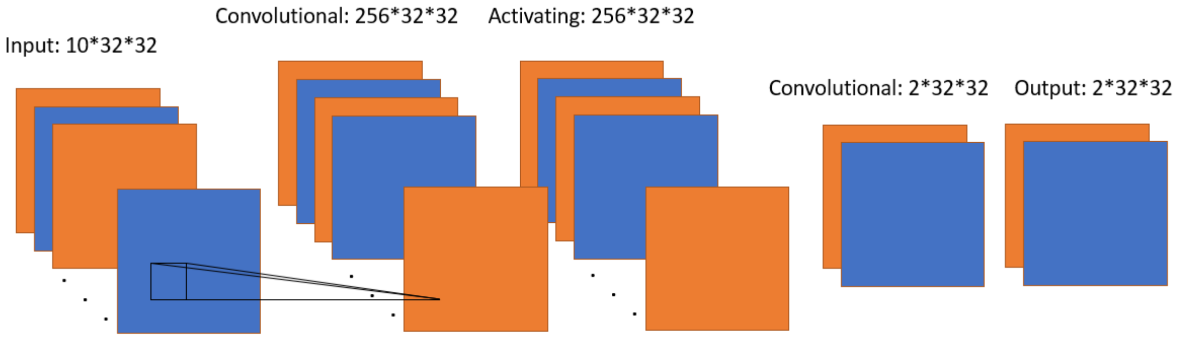


Figure 3: CNN Model

3.4 Evaluation Values

In this report, there are three values which have been used to analyze the performance and help to set the value of parameters. The Mean squared error (MSE) measures the average of the squares of the errors which is a risk function, corresponding to the expected value of the squared error loss. Considered the situation that the MSE may too big to show the performance, Root Mean Square Error (RMSE) is also be calculated. It gives a relatively high weight to large errors. Besides, the L1Loss in pytorch is the value of Mean Absolute Error (MAE).

$$MSE = \frac{\sum_{i=1}^{n}(\bar{y_i} - y_i)^2}{n} \tag{1}$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(\bar{y_i} - y_i)^2}{n}} \tag{2}$$

$$L1Loss(MAE) = \frac{1}{n}\sum_{i=1}^{n}|y_i - \bar{y_i}| \tag{3}$$

4 Solutions

4.1 Feature Extracting

The data is given in two h5 files containing date (7720,), crowd flows (7720, 2, 32, 32), and meteorology: temperature(7720,) weather(7720, 17, 1) wind-speed(7720,). Therefore, data should be reorganized to the dimensions (m, 32, 32, n), where m is the size of the training set and n is the number of features. For temperature, weather, and wind-speed, which are recorded once in one time slot, they are extended into the size of 32*32. In these processes, *numpy* is used to shape the array. Besides, data lacking features are deleted before training.

4.2 Module

As mentioned, **CNN** is used to build the model. As the requirement of output, pooling layers and connectivity layers are unsuitable. Pooling layer extracts the comprehensive features, however, the features of each grid should be personalized and preserved in this model. As the requirement of output (32*32*2), the connectivity layer is not applied. In addition, the module is build using *pytorch*.

4.3 Loss Function

MSE is used as the loss function. It is the most commonly used loss function in regression. As mentioned in 3.4, *MSE* is the average of squared distances between our target variable and predicted values, and therefore it is sensitive to errors bigger than 1, and gives a larger penalty to them. For the crowd, if the difference between observed value and prediction is smaller than 1, the error can be tolerated. Therefore, *MSE* is able to help select variables that are profitable to constrained functions constringency in this model.

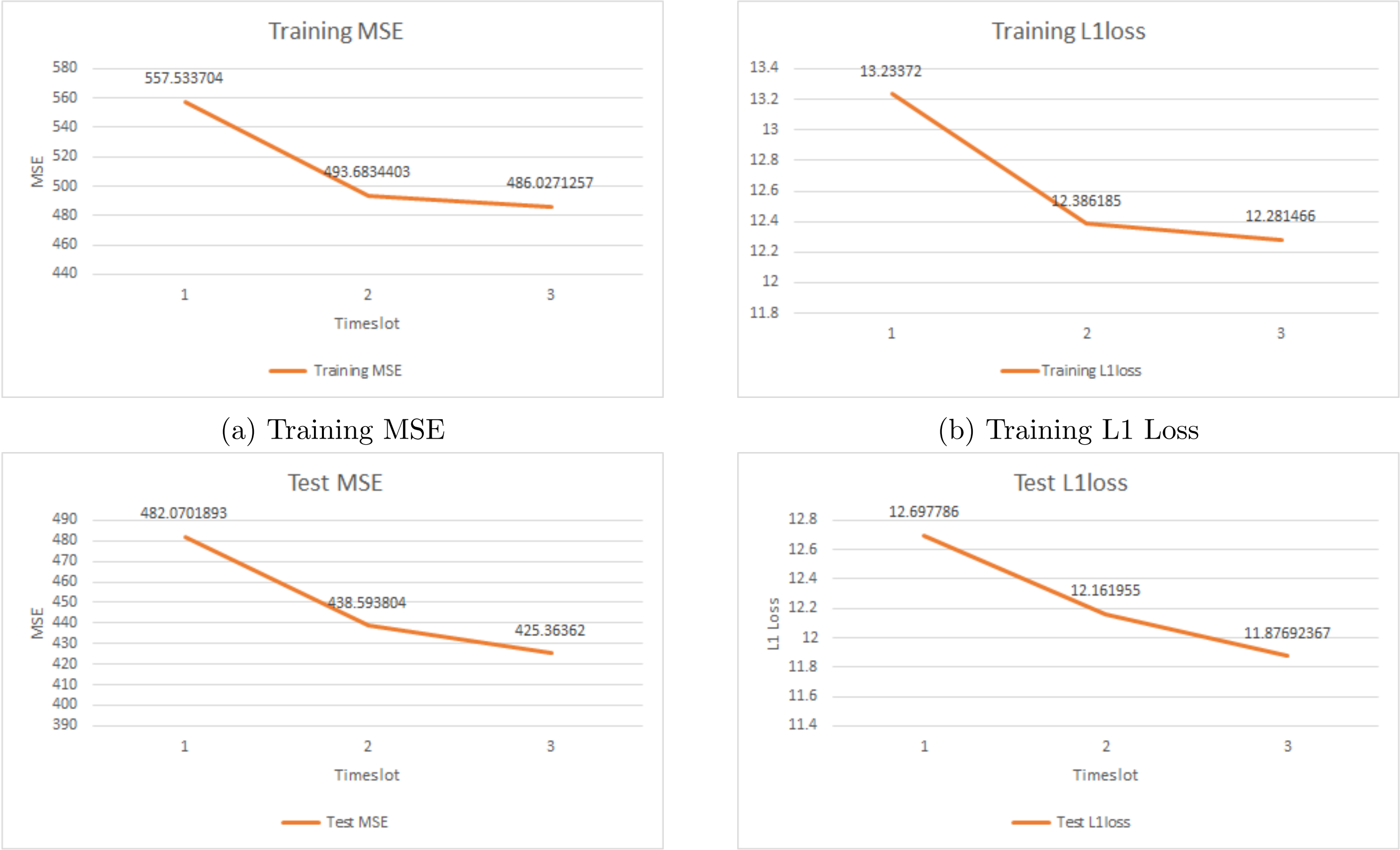
4.4 Optimization

Having built the model, 80% of data is set as training data, and 20% is set as test data. After each running, the results including time cost and the evaluation values are used for optimization. During the optimization processes, different learning rates, inputs, batch sizes, and other parameters are tried. The model with the least time consumption and smallest evaluation values is selected.

5 Performance evaluation and discussions

5.1 Impact of Different Configuration

Closeness: We tried different number of previous timeslots as the input. The Fig.4 below shows the result. It is shown that when the number of previous timeslots used as the input is 3, the training results and testing results are both better.



(a) Training MSE

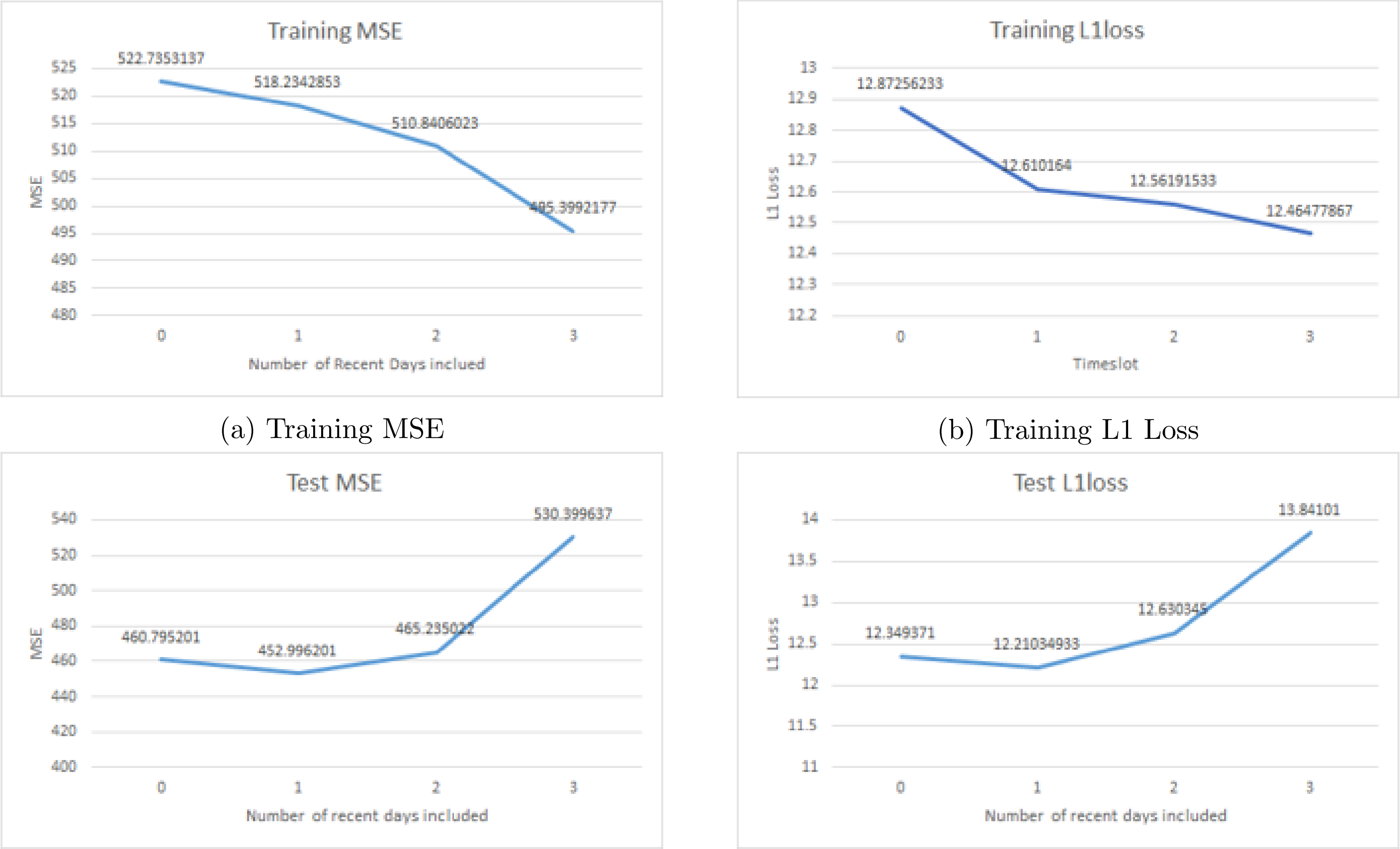
(b) Training L1 Loss

(c) Testing MSE

(d) Testing L1 Loss

Figure 4: The Impact of Closeness

Period: We tested the impact of using the different number of previous days' the same time's timeslots as the input. The Fig.5 below shows the result. It is shown that when the number is 1, the testing results is the best.



(a) Training MSE

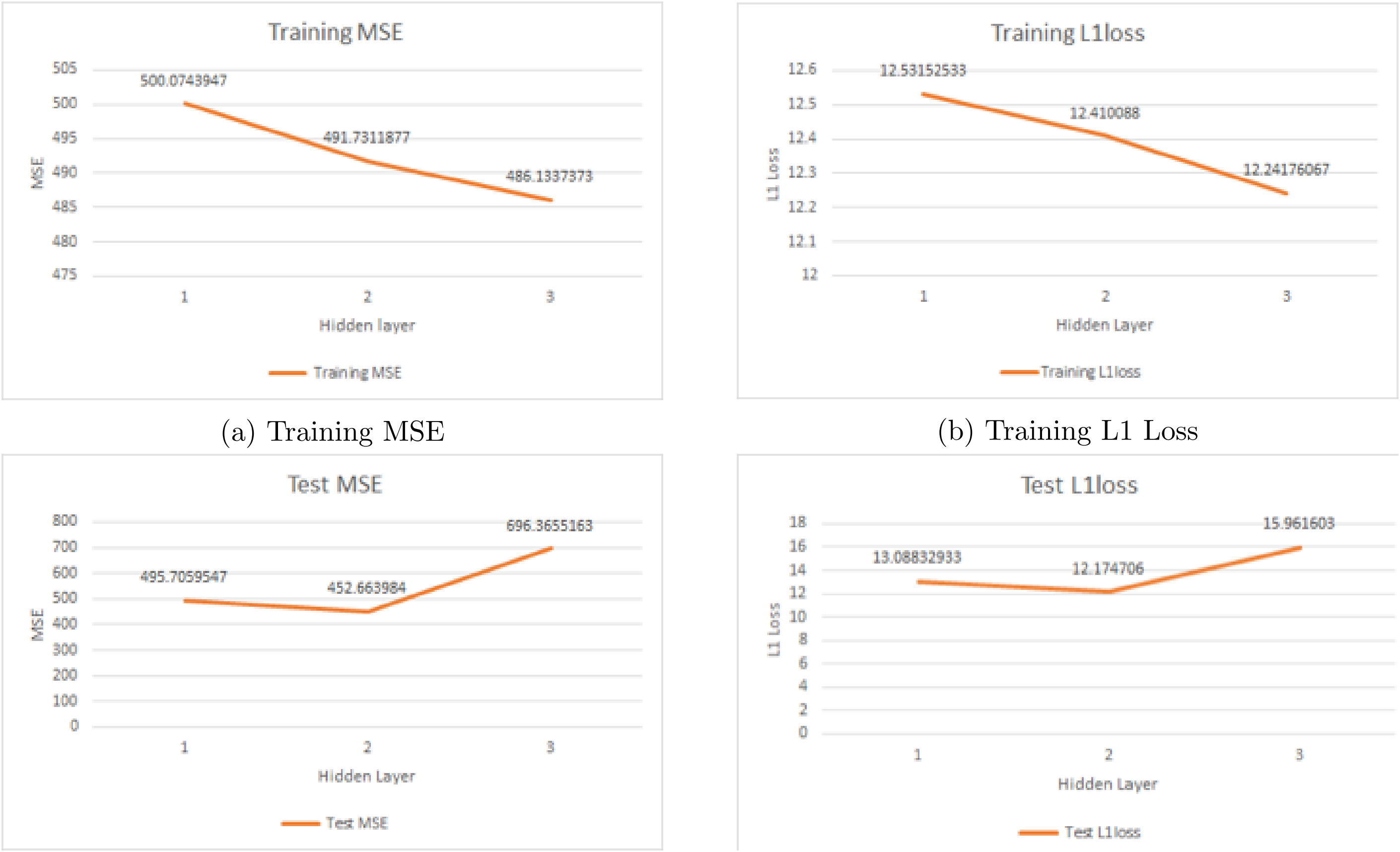
(b) Training L1 Loss

(c) Testing MSE

(d) Testing L1 Loss

Figure 5: The Impact of Period

Number of Hidden Layers: We tested the impact of different number of hidden layers. The Fig.6 below shows the result. It is shown that with the increase of the number of hidden layers, the training results become much better. However, 2 hidden layers achieve the best result in testing MSE Loss and testing L1 Loss. A possible reason may be that the over-fitting problem occurs when there are 3 hidden layers.



(a) Training MSE

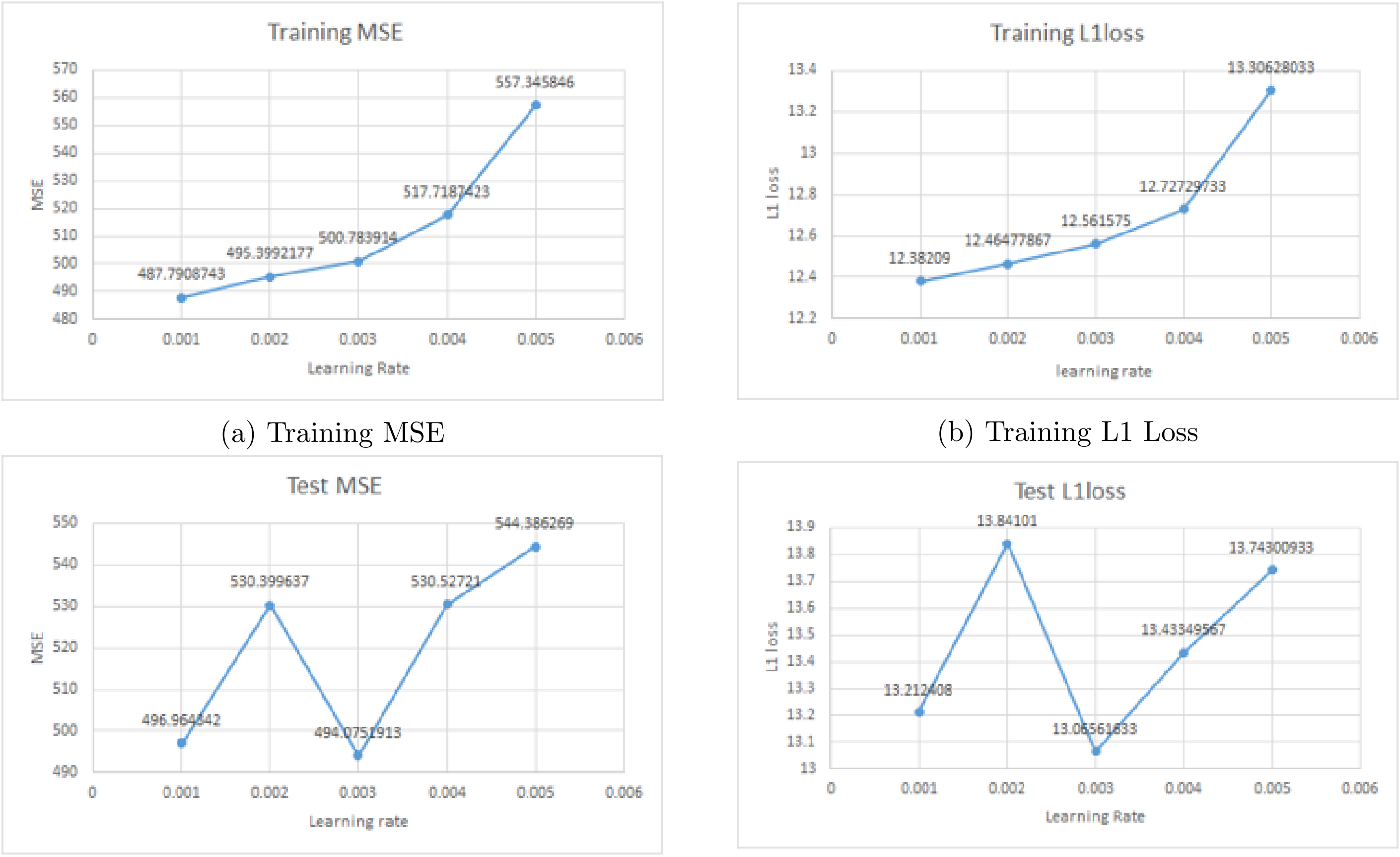
(b) Training L1 Loss

(c) Testing MSE

(d) Testing L1 Loss

Figure 6: The Impact of Hidden Layers

Learning Rate: We tested the impact of different learning rate. The Fig.7 below shows the result. It is shown that with the increase of learning rate, the training result becomes worse. As for the testing result, 0.001 and 0.003 perform well. Considering both training result and testing result, finally we choose 0.001 as our model's learning rate.



(a) Training MSE

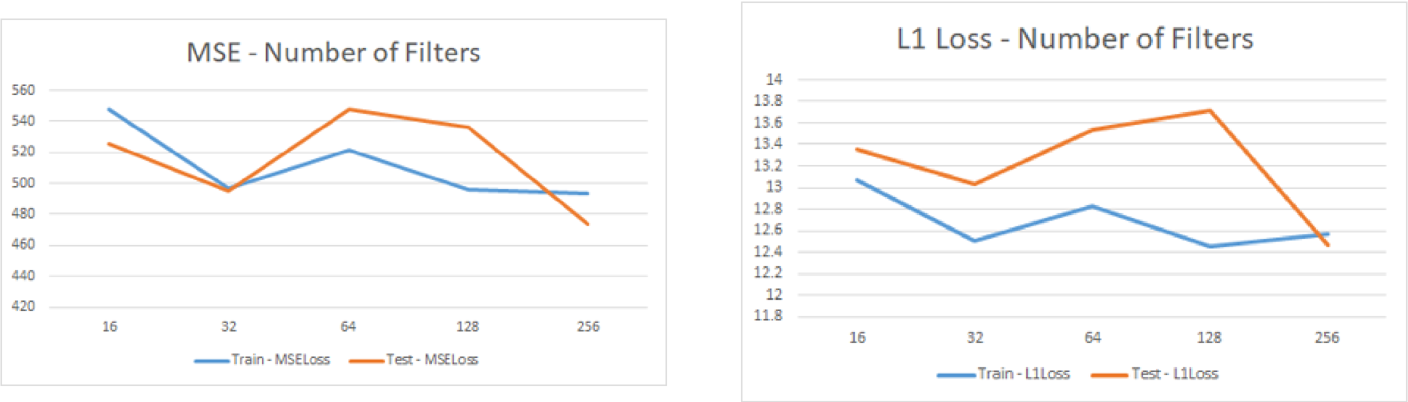
(b) Training L1 Loss

(c) Testing MSE

(d) Testing L1 Loss

Figure 7: The Impact of Learning Rate

Number of Filters: We tested the impact of different number of filters. The Fig.8 below shows the result. It is shown that 256 filters perform both well on training result and testing result.

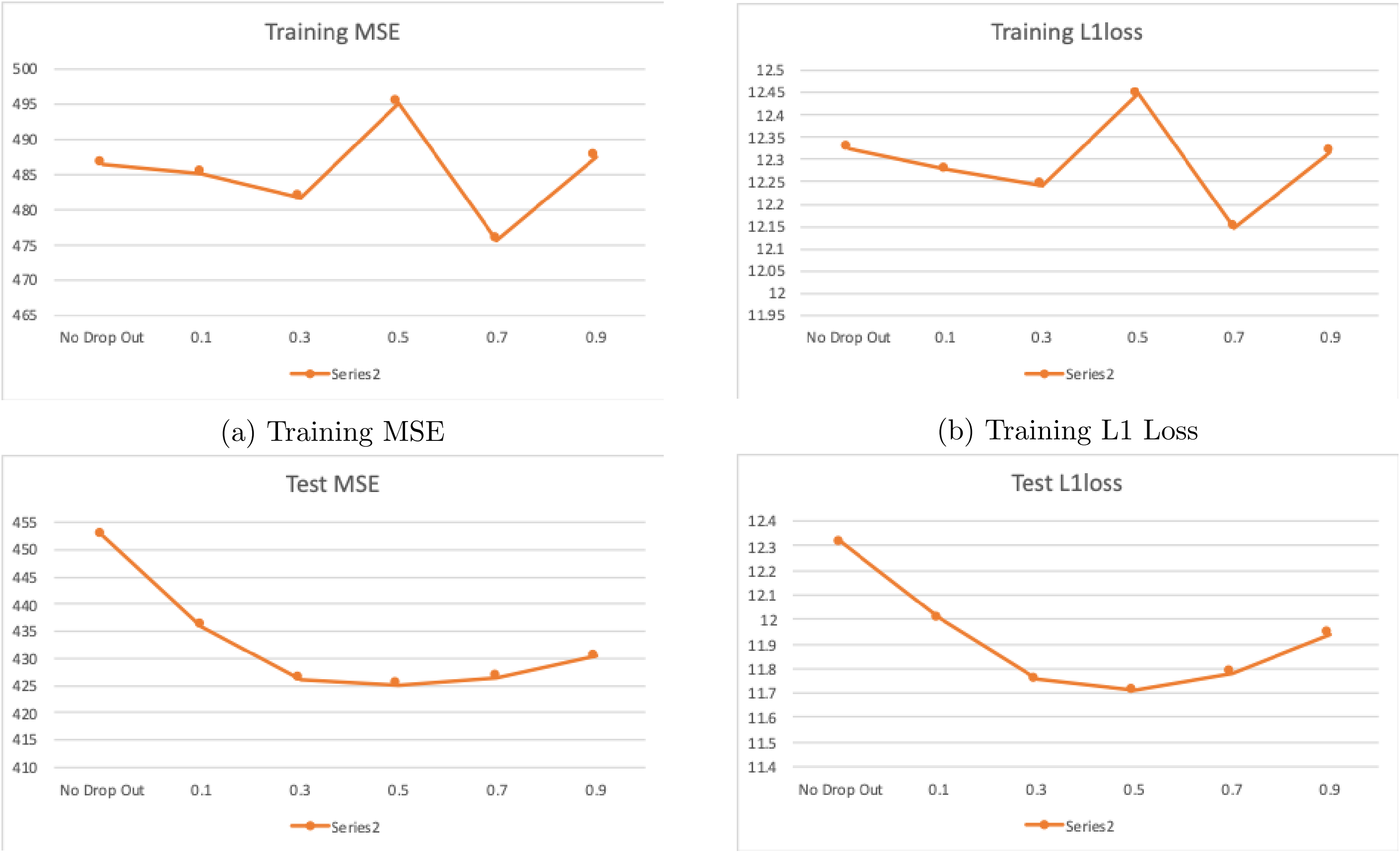


(a) MSE

(b) L1 Loss

Figure 8: The Impact of The Number of Filters

Drop Out: Drop out means how many percentages of factors should be disposed. For the drop out factor, we have tested 0.1, 0.3, 0.5, 0.7, 0.9 and no drop out (0.0). Since difference occurs between the training MSE and Test MSE, we finally adopted drop out equals to 0.5 which has the best performance in test MSE.



(a) Training MSE

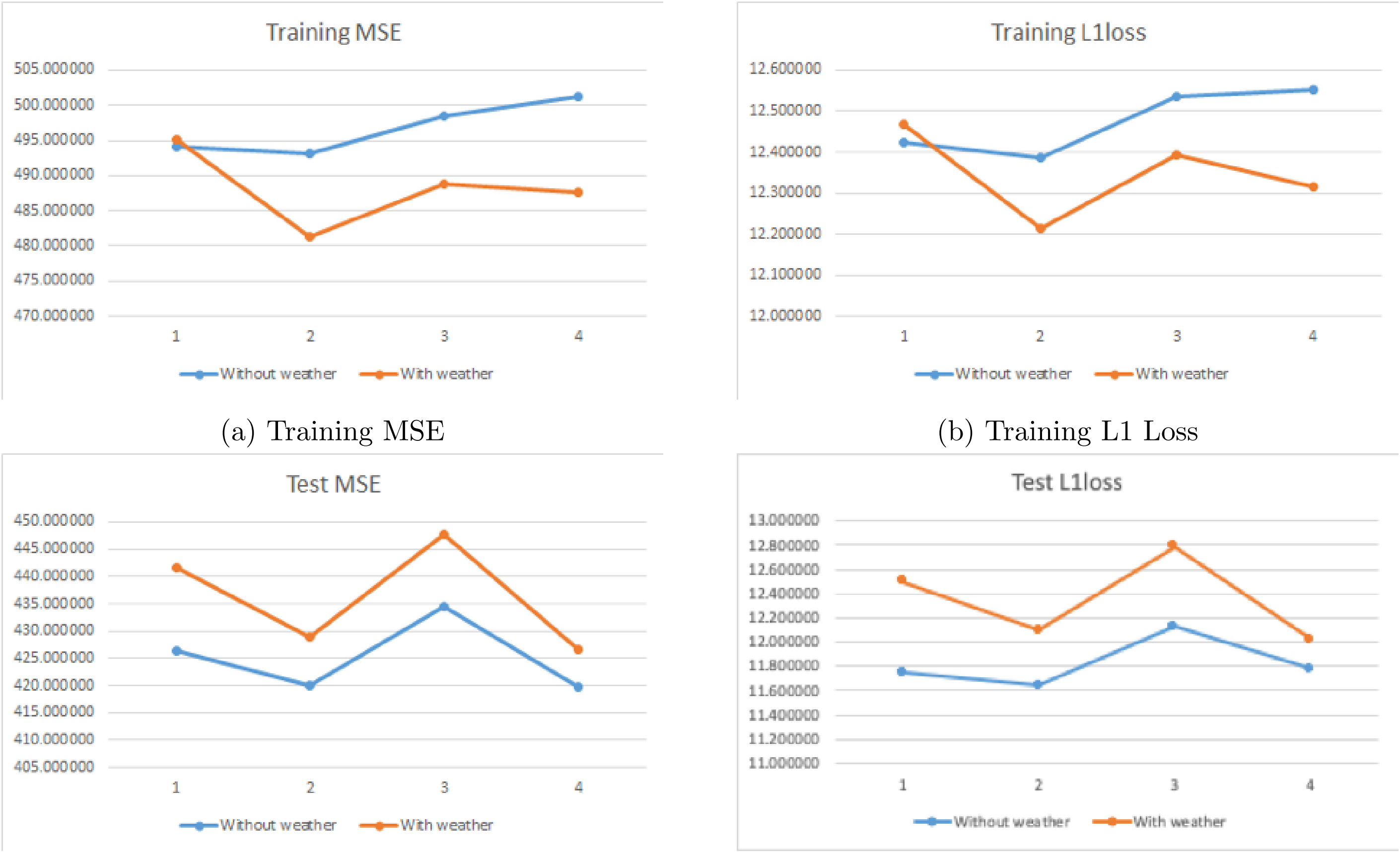
(b) Training L1 Loss

(c) Testing MSE

(d) Testing L1 Loss

Figure 9: The Impact of Drop Out

External Influence (Weather): With drop equals to 0.5, we tested our model four times with either weather factor exists or not. The result shows that the model without weather has better performance than the model with weather. Therefore, our final model adopts the version without weather.



(a) Training MSE

(b) Training L1 Loss

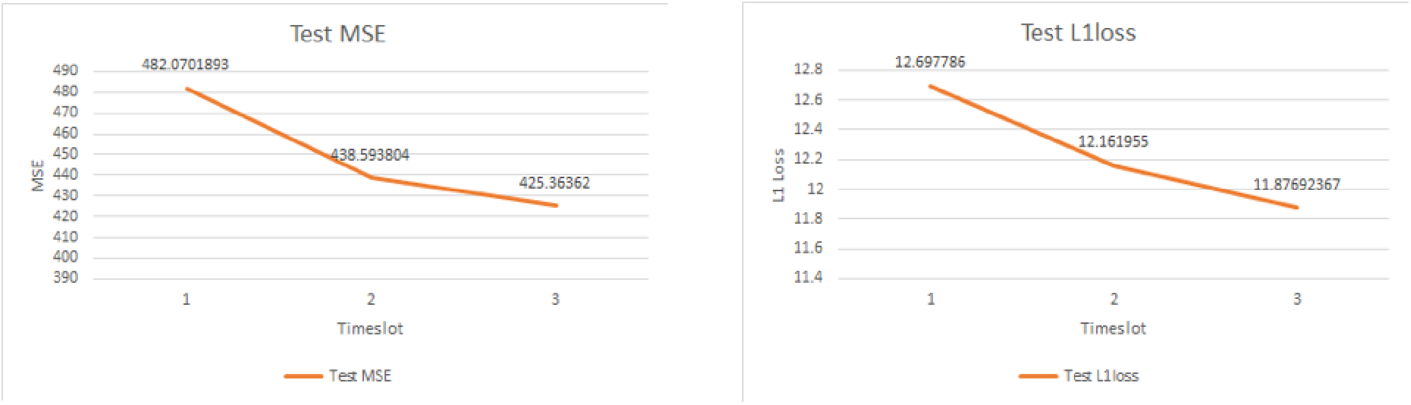
(c) Testing MSE

(d) Testing L1 Loss

Figure 10: The Impact of Weather Factor

5.2 The Input Data Contributes Most

The number of timeslots used contributes most to our model accuracy rate. When the timeslots increased from 1 to 3, the test MSE of our model dropped down sharply. One possible reason could be that most regions in the city have more or less closeness.



(a) MSE

(b) L1 Loss

Figure 11: Timeslots Contributes Most

5.3 Final Model

With either hidden layer equals to 1 or 2, we developed two models where the first one is more time efficient (training time 570 seconds, test RMSE 20.198342) and the second one has better performance (training time 4915 seconds, test RMSE 19.296269). The values below are the average value after running the model 5 times.

5.3.1 Time Efficient Model

Running Parameters:
Kernel Size = 3
Epoches = 50
Batch Size = 64
Number of Filters = 256
Dropout = 0.5
Learning Rate=0.001
Hidden Layers = 1

Performance:
Train MSE: 456.259246
Train RMSE: 21.360226
Train L1 Loss: 11.8522027
Test MSE: 407.9730357
Test RMSE: 20.198342
Test L1 Loss: 11.4629663

Training Time:
11.4 s/epoch * 50 epoch = 570 seconds

Hardware Environment:
GPU GTX 1080Ti, RAM 64G, CPU i9-7900X, Windows10

5.3.2 Better Performance Model

Running Parameters:
Kernel Size = 3
Epoches = 50
Batch Size = 64
Number of Filters = 256
Dropout = 0.7
Learning Rate=0.001
Hidden Layers = 2

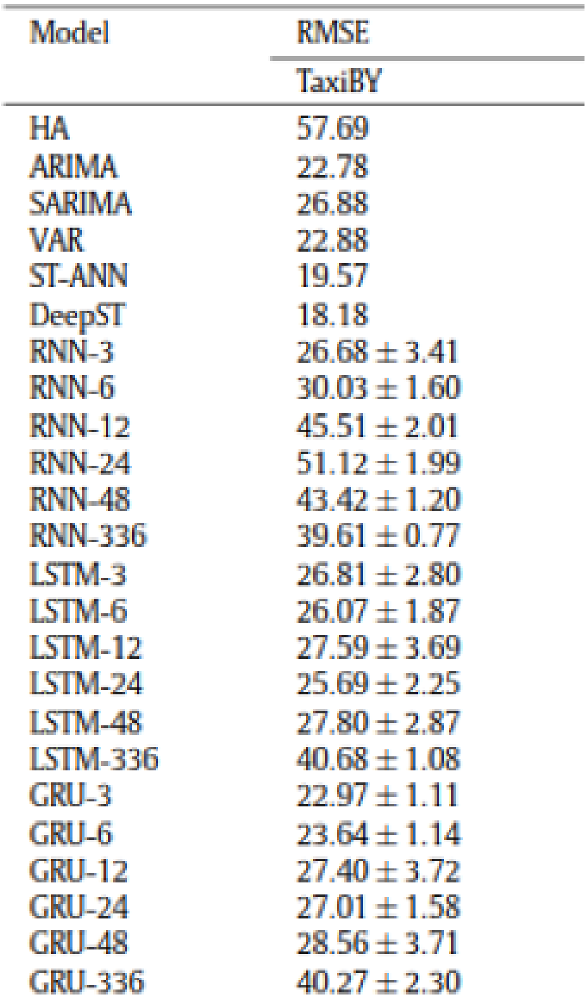
Performance:
Train MSE: 428.087264
Train RMSE: 20.69027
Train L1 Loss: 11.517178
Test MSE: 372.346
Test RMSE: 19.296269
Test L1 Loss: 11.001296

Training Time:
98.3 s/epoch * 50 epoch = 4915 seconds

Hardware Environment:
GPU GTX 1080Ti, RAM 64G, CPU i9-7900X, Windows10

5.4 Comparison with Different Models

Fig.12 shows several popular models and their RMSEs on the same topic [1]. Note that the input data for these models and our model are exactly the same. Comparing with our best performance model with RMSE 19.30, only model DeepDT has lower RMSE which is 18.18. Our model exceeds all the other models.



Model	RMSE
	TaxiBY
HA	57.69
ARIMA	22.78
SARIMA	26.88
VAR	22.88
ST-ANN	19.57
DeepST	18.18
RNN-3	26.68 ± 3.41
RNN-6	30.03 ± 1.60
RNN-12	45.51 ± 2.01
RNN-24	51.12 ± 1.99
RNN-48	43.42 ± 1.20
RNN-336	39.61 ± 0.77
LSTM-3	26.81 ± 2.80
LSTM-6	26.07 ± 1.87
LSTM-12	27.59 ± 3.69
LSTM-24	25.69 ± 2.25
LSTM-48	27.80 ± 2.87
LSTM-336	40.68 ± 1.08
GRU-3	22.97 ± 1.11
GRU-6	23.64 ± 1.14
GRU-12	27.40 ± 3.72
GRU-24	27.01 ± 1.58
GRU-48	28.56 ± 3.71
GRU-336	40.27 ± 2.30

Figure 12: Comparison with Different Models
[1]

6 Future development

As mentioned in 5.4, the final model performs satisfactorily. However, there is room for improvement.

6.1 Feature Selection

It is observed that the predictions of flows in weekdays are much more accurate than those on weekends and special days. Therefore, it is supposed that the factor whether the time slot is from weekend or special day matters. However, as the data set contains data only in half a year, it is hard to train the model with this factor. In future improvement, this factor can be added if a larger data set is collected.

The result performs without improvement when the weather is added in directly as new channels. In future development, weather can be considered in another way. One proposal is to build a model to analyze the relationship between weather and flows and add the results to the present model.

6.2 Input Sequence

As the order of input matters in **CNN**. Neighbor channels influence each other. Thus different sequence of inputs 3.2 can be arranged and tried to train the model.

6.3 Distant Spatial Dependency

Consider distant spatial dependency. Apart from the nearby feature, the crowd flows may also be affected by the flows of distant regions. For example, people living far away from their office usually take the subway to work, thus the outflow of the distant area directly indicate the inflow of the office area.

References

[1] J. Zhang, Y. Zheng, D. Qi, R. Li, X. Yi, and T. Li, "Predicting citywide crowd flows using deep spatio-temporal residual networks," *Artificial Intelligence*, vol. 259, pp. 147–166, 2018.

A Documented source files and user manual

B Contributions of team members

The contributions of team members are equal:
CHENG Yiran: Analyzing the input data, Coding, Testing, Debugging, Writing the report, Preparing the presentation, Searching the related papers and information
HAO Shiqi: Analyzing the input data, Coding, Testing, Debugging, Writing the report, Preparing the presentation, Searching the related papers and information
JIANG Yuxin: Analyzing the input data, Coding, Testing, Debugging, Writing the report, Preparing the presentation, Searching the related papers and information
WANG Bokang: Analyzing the input data, Coding, Testing, Debugging, Writing the report, Preparing the presentation, Searching the related papers and information
WANG Jiashuo: Analyzing the input data, Coding, Testing, Debugging, Writing the report, Preparing the presentation, Searching the related papers and information
YU Jing: Analyzing the input data, Coding, Testing, Debugging, Writing the report, Preparing the presentation, Searching the related papers and information