Project Milestone

Panda Xu px48 **Scarlett Huang** sh2557 **Zihan Zhang** zz698

1 Motivation

With the advent of the mobile Internet era, all mobile device holders can become portrayers of traffic conditions and road capacity. If it is possible to have a more accurate prediction of traffic conditions based on real-time and historical traffic information, it will undoubtedly play a vital role in travel decision-making and alleviating urban congestion.

However, it is very difficult to predict future road conditions, which are affected by many factors such as time period, road capacity, the downstream topology of road networks, navigation traffic, and sudden road conditions.

The goal of our project is to accurately predict the traffic conditions (ie, unblocked, eased, and congested) in a certain period of time in the future based on real-time and historical road condition characteristics of road segments, basic road attributes, and road network topology relationship diagrams to help drivers to optimize route plans as well as city planning.

The project is an application based on real world data and may be used on improving the algorithm of Didi.

2 Method

Most of the prior works use probabilistic models to predict the traffic conditions, which are mainly suitable for traditional data with small sample size and simple data structure. However, the performances of traditional prediction methods for big data problems with large amounts of data and high complexity are severely restricted by data noise and influences of emergencies. Our project plans to apply machine learning and deep learning methods to solve this problem.

First, we will conduct data preprocessing and feature engineering such as normalization, feature screening and dimensionality reduction on road sections to be predicted. e.g. As future road conditions are affected by many factors (time period, road capacity, etc.), we can construct multi-dimensional feature vectors.

Second, we will try a variety of machine learning classification algorithms, including SVM, Naive Bayes, Decision Tree, Xgboost, KNN, Logistic Regression to build classifiers separately.

Third, we will also try network models such as CNN and RNN. Considering the complexity of the spatial correlation of traffic status, we can use graph convolution networks to process spatial information.

Fourth, we will apply ensemble learning methods, including boosting and bagging, to traverse different model combinations in turn to find the best ensemble model to achieve accurate short-term prediction of the road network traffic status.

Fifth, we will run experiments and try to improve model structure to continue optimizing model performance. e.g. We can consider using attention mechanisms to improve the model.

Sixth, we will also look up more relevant research papers to find better models to achieve better prediction. e.g. DCRNNs is an Encoder-Decoder model based on graph convolution, which performs well on traffic status prediction [1].

3 Preliminary experiments

3.1 Get Training Data and Test Data

We signed up for the road condition spatio-temporal prediction competition organized by China Computer Society and Didi Travel on DataFountain. By submitting an application for data acquisition, the original data set was downloaded on the Gaia Data Open Project website. This competition provides real-time and historical road condition information of Xi'an, as well as road attributes and road network topology information on the Didi platform from July 1, 2019 to July 31, 2019.

3.2 Data Preprocessing

The training data set consists of three parts. The first part is traffic data, including road segment id, road condition status, current time, and future time, the recent five time traffic characteristics and the five traffic characteristics of the same period in the past four weeks.

```
353495 1 236 245;232:29.80,32.40,1,4 233:31.60,32.20,1,2 234:20.00,21.90,2,2
235:22.20,25.90,2,5 236:21.30,26.30,2,4;245:30.00,32.70,0,9 246:30.00,36.10,0,10
247:27.40,35.20,1,12 248:26.90,35.70,1,10 249:28.90,37.00,1,9;245:36.10,37.30,1,7
246:29.30,38.50,1,7 247:27.70,39.70,1,6 248:28.60,40.20,1,3
249:29.60,38.70,1,4;245:30.40,40.10,1,6 246:32.30,40.10,1,6 247:30.60,41.10,1,5
248:29.60,39.20,1,4 249:28.00,37.90,1,4;245:28.30,38.40,1,7 246:28.20,39.40,1,6
247:28.80,35.10,1,3 248:30.00,35.60,1,4 249:29.40,37.20,1,5
```

Figure 1: Traffic

The second part is the road attributes, which respectively represent the link id, link length, traffic direction, function registration, speed limit level, number of lanes, speed limit, classification and width.

```
0       19      1       5       7       1       4.166667        5       30
1       19      1       5       7       1       4.166667        5       30
2       16      1       5       7       1       4.166667        5       30
```

Figure 2: Attribute

The third part is the road network topology, the key is the upstream link id, and the value is the downstream link id.

```
611897  630844,611898,611691
611704  612102,611703
611656  611318,611657,611315
```

Figure 3: Topo

Use python to read the .txt file, remove the punctuation marks of the first part of the traffic data, extract the road condition status as Ytrain, and the rest as Xtrain. At the same time, in order to facilitate the analysis of each feature, save all 129 features in a piece of data as lists. For the second part of data, save it as a dataframe, and for the third part of data, save it as a dictionary.

3.3 Selecting, Importing, and Splitting the Training and Testing Dataset

The attr.csv, topo.csv, and traffic.csv files were obtained from the previous preprocessing step. Based on our understanding of the relevant factor of traffic conditions, we decided to first analyze the traffic conditions for all road segments in traffic.csv, which consists of the speeds of passing vehicles, road condition status, number of vehicles passing through, etc.

```
1 import numpy as np
2 import pandas as pd
3 from sklearn.feature_extraction.text import CountVectorizer
```

```
dataset = pd.read_csv('preprocessing/traffic.csv')[:10000]
target_col = dataset["label"]
train = dataset.sample(frac = 0.7)
dev = dataset.drop(train.index)
print(train.shape)
print(dev.shape)
train_Y = train["label"].to_numpy()
dev_Y = dev["label"].to_numpy()
train_X = train.drop(columns=['label']).to_numpy()
dev_X = dev.drop(columns=['label']).to_numpy()
print(train_Y.shape)
print(dev_Y.shape)
```

A support vector machine (SVM) was used to train and predict the traffic condition. In order to evaluate the prediction accuracy, 70 % of the whole dataset was divided into the training set and 30 % of it into the test set.

```
from sklearn.svm import LinearSVC
from sklearn.metrics import f1_score
f1_list = []
ci_list = [0.01, 0.1, 1, 10, 100]
for ci in ci_list:
    lsvc = LinearSVC(C = ci, max_iter = 100000)
    lsvc.fit(train_X, train_Y)
    pred_dev = lsvc.predict(dev_X)
    y_true = dev_Y
    score = f1_score(y_true, pred_dev, average=None)
    f1_list.append(score[0])
print(f1_list)
```

In order to find out the best cost parameter C, a list of five Cs (0.01, 0.1, 1, 10, 100) were selected to fit into the SVM model. We used the F1 score to evaluate the prediction accuracy with different Cs, and the result was obtained as follows:

[0.9102470930232558, 0.7044614246980527, 0.9051010587102984, 0.9051823416506718, 0.8238084968729782] We found out that the highest f1 score, 0.91 was obtained when c = 0.01.

4 Future Work

First, we plan to use more ML models such as Naive Bayes, neural networks, decision trees and so on to model and predict data respectively. In this period of work, the main tasks are focused on data preprocessing and discussing modeling methods. due to the large amount of data (about 180,000 pieces), the train and dev set used this time are relatively small. In future work, we will use all the data for training and prediction. At the same time, we will continue to improve the SVM model, use more c for selection, and adjust more gamma.

In the follow-up work, we plan to analyze the five attributes of the characteristics at different times in the traffic data separately to estimate the importance of different attributes. Next, the attributes of high importance will be combined, and then the overall data would be predicted. At the same time, we plan to consult the data, theoretically analyze the impact of road attributes and network topology on the road condition, and use attribute data and topology data for analysis and verification. Finally, we will find the proper model with the best prediction score, make predictions on the test set, and submit it to the competition website for scoring.

References

[1] Li, Yaguang, et al. (2017) Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. *arXiv preprint arXiv*:1707.01926.