# Machine-learning Based Model for Short-term Traffic Prediction

**Yuhao Chen**
Department of MAE
yuc043@ucsd.edu

**Tianmu Wang**
Department of MAE
tiw028@ucsd.edu

**Haonan Peng**
Department of ECE
hap045@ucsd.edu

## Abstract

With the expansion of metropolises and the high demand of vehicles, improving the capacity of road networks is no longer simply planning more roads and more lanes. Besides, the increasing number of roads is unavoidably raising the complexity of road networks which brings difficulty in monitoring and maintaining the traffic environment. Under that urgent demand, the Intelligent Transportation System (ITS) attracted increasing attention and developed fast in recent years. Among this gigantic system, short term traffic prediction is a vital branch, especially in real-time route guidance. In this project, our goal is to propose a robust and accurate machine-learning based model for short-term traffic prediction.

## 1 Introduction

Accurate and timely traffic flow information is strongly needed. The goal of this project is to do short-term prediction about the traffic flow. Traffic flow prediction heavily depends on historical and real-time traffic date, which is collected by official institutions. This work's data for training is obtained from the PeMS, which shows recorded traffic flow data for the Freeway SR52-E in District 11 (San Diego) for the past few years. In this project, the Gradient Boosting Decision Tree (GBDT), a popular machine learning algorithm in classification and regression field and needs to scan all the data instances to estimate the information gain of all possible split points [1], would be used as Liu [2] did. In this case, the SR52-E connects the University City to the other area, so the prediction of VMT could be used to estimate the rough lane occupancy, which might provide the information to help to schedule the maintenance plan.

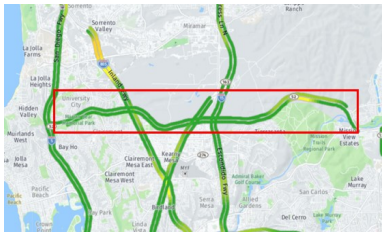


Figure 1: Location of the example freeway

## 2 Methodology

### 2.1 Processing

1) Training dataset obtain:

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

The raw training dataset is the traffic flow data of Freeway SR52-E in 2021, which has been download from PeMS, who contains the following features: Month, Date, Hour, Last Hour Vehicle Hours Traveled (VHT), number of Lane Points, percentage of the observed vehicles, and the Last Hour value Vehicle Miles Traveled (VMT). In conclusion, more than 7,000 data points with 7 features are used for building the model, and the prediction should be the VMT for the next hour.

2) Split of training dataset:

Since pruning of decision trees requires validation data, the training dataset should be shuffled to remove linearity, and chosen 30 percent of the training dataset randomly by the sklearn function $train\_test\_split$.

3) Testing dataset obtain:

The raw testing dataset is the traffic flow data of the same freeway in 2022 which would not be exactly the same as the 2021.

### 2.2 GBDT Modeling

The GBDT is a Ensemble Model, who is to combine several models (commonly known as weak estimator) according to a certain strategy to jointly complete a task, then a specific combination strategy would help the ensemble model reduce the bias or variance of predictions. It is also known as MART(Multiple Additive Regression Tree), which is for classification and regression problems. The algorithm could train the newly added Weak Classifier according to the negative gradient information from current loss function, then combine the well-trained weak classifier with the existence model by accumulation. The GBDT is using the boosting strategy.

**Boosting Strategy** Boosting is an ensemble learning method that combines a set of weak learners into a strong learner to minimize training errors. In boosting, a random sample of data is selected, fitted with a model and then trained sequentially—that is, each model tries to compensate for the weaknesses of its predecessor. With each iteration, the weak rules from each individual classifier are combined to form one, strong prediction rule. Boosting models combining the basic models by concatenation. The idea of this type of model is that one basic model can make imperfect predictions, so more models could be applied to polish the imperfect parts. There are three such algorithms: AdaBoost (Adaptive Boosting) algorithm; Gradient Boosting algorithm; XGB (Extreme Gradient Boosting) algorithm. And the actual structure is showing in figure 3.

**CART** The decision tree algorithm is a typical classification method to approximate the value of a discrete function. First, it would process the data using an inductive algorithm to generate the readable rules and decision trees. Then, it would use the decisions to analyze the new data, so as to achieve the goal of classification or regression. Thus, essentially, the decision tree is the process of classifying data through a series of rules. The classic algorithm of the decision tree are ID3, C4.5 and CART. GBDT is a boosting combination of k's CART, so CART is the basic model. It is a non-parametric decision tree learning technique that produces either classification or regression trees, depending on whether the dependent variable is categorical or numerical. In such regression problem, the binary tree should be used to complete linear regression tasks. To construct the tree, the algorithm will first generate some split filters according to the Gini impurity, then the variables that satisfies the filters would be dropped to the left branch while the rest in the right branch. The split would keep processing until the stop criteria reached (like the maximum tree depth). For the regression tree, the Gini impurity would be replaced by the variance reduction, which is used to increase the precision of the estimations by:

$$VE = \frac{1}{N}\sum_{n=1}^{N} Var_n - (\frac{length_{left}}{length} \times Var_{n_{left}} + \frac{length_{right}}{length} \times Var_{n_{right}})$$

Then after building the model, the validation dataset would be used to test the model and the tree would be pruned according to reduce error pruning (REP), so the model would eliminate the over-fitting branches.

**Gradient and Loss function** Gradient-boosted decision trees are a popular method for solving prediction problems in both classification and regression domains. The approach improves the learning process by simplifying the objective and reducing the number of iterations to get to a sufficiently optimal solution. Gradient-boosted models have proven themselves time and again in various competitions grading on both accuracy and efficiency, making them a fundamental component in the data scientist's tool kit. The gradient is used to accelerate the process of minimizing the loss function, which is the criterion how the algorithm judges the quality of training. In this case, the Huber loss, a loss function used in robust regression and it is less sensitive to outliers in data than the squared error loss, and the relative square loss gradient applied to construct the boosting model. And in this case the sigma is equal to 1.

$$L_\delta(y, f(x)) = \begin{cases} \frac{1}{2}(y - f(x))^2 & , \text{if } |y - f(x)| < \delta \\ \delta \times (|y - f(x)| - \frac{1}{2}\delta) & , otherwise \end{cases}$$

$$Gradient_\delta(y, f(x)) = \begin{cases} y - f(x) & , \text{if } |y - f(x)| < \delta \\ \delta \times sign(y - f(x)) & , otherwise \end{cases}$$

## 2.3 Prediction

Prediction in machine learning refers to the output of an algorithm after it has been trained on a historical dataset and applied to new data when forecasting the likelihood of a particular outcome. The result is critical for the machine learning models. The quality of a model depends on the predicted value. Sometimes the better a model fits the training data, the worse the result may be. Thus after training and pruning the model, we need to test its accuracy and performance if it could meet the desired effect. Then here we use another data to generate the prediction value. Here, in order to verify the correction of the model and avoid overfitting, we randomly selected one day as the testing data. The dataset for testing the Caltrans Performance Measurement System (PeMS). We see how accurate the result is by comparing the actual values and the prediction values through the model:

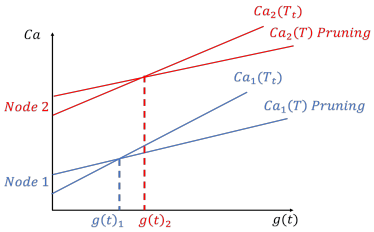



Figure 2: Error Difference Before and After Pruning

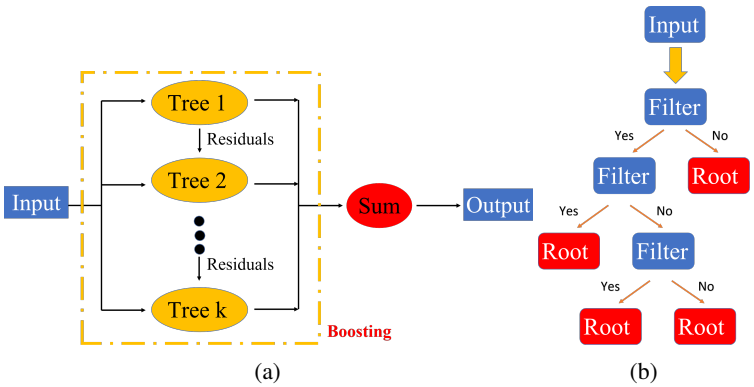



Figure 3: GBDT Architecture: a) Boosting model, b) CART model

## 3 Data Description

The training data we used in the study is downloaded from the Caltrans Performance Measurement System (PeMS) (`https://pems.dot.ca.gov/`), which is an open-access, real-time database collecting the traffic data of the freeway system across all major metropolitan areas of the State of California from over 39,000 individual detectors. We collected the traffic data of the State Route 52 in San Diego County, California from January 1 to April 30, 2022. The selected segment is a major east-west route with entire length 14.8 miles monitored by 44 detectors. In our study, the data we collected is grouped by: hour (represented in 24-hour clock), vehicle hours traveled, vehicle miles traveled (the product of vehicle hours traveled and entire length of route), lane point, and the percentage of observed vehicle. The detailed statistic sample is shown in Table 1. The traffic data of January, February, and April is selected to train our model, while the validation data is the data collected on March. It should be mentioned we selected the data on March 1st to test our model at the current stage.

Table 1: Statistics of traffic data sample

| Month | Day | Hour | Next VMT | Previous VHT | #Lane Points | %Observed | Previous VMT |
|---|---|---|---|---|---|---|---|
| 4 | 1 | 0 | 5,260.90 | 79.10 | 480 | 55.6 | 6401 |
| 4 | 1 | 1 | 3,916.50 | 59.30 | 480 | 55 | 5260.9 |
| 4 | 1 | 2 | 3,683.00 | 55.60 | 480 | 55 | 3916.5 |
| 4 | 1 | 3 | 4,486.30 | 67.90 | 480 | 55 | 3683 |
| 4 | 1 | 4 | 7,591.00 | 113.50 | 480 | 55 | 4486.3 |

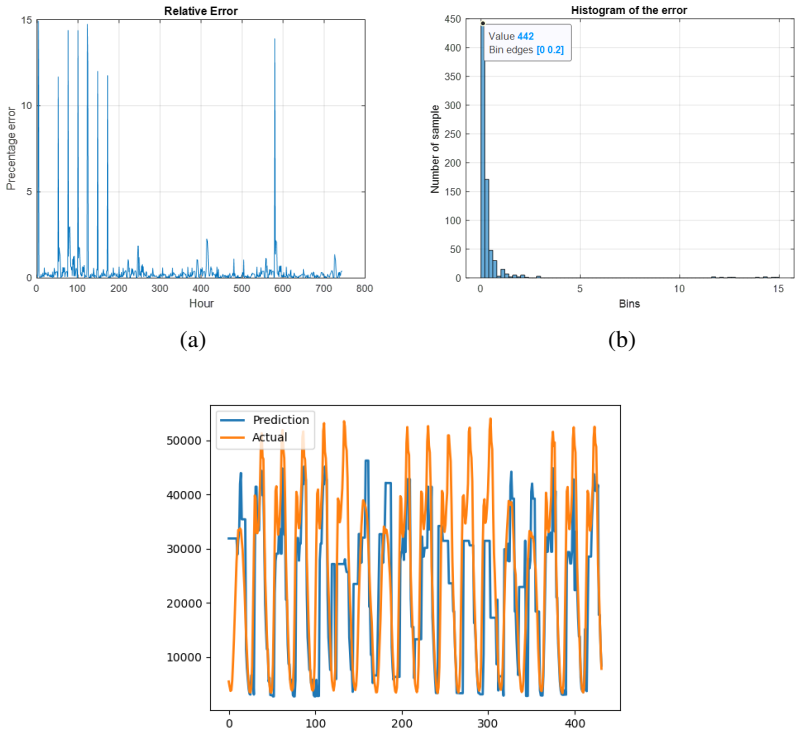
## 4 Experiment

The general training process is as follows. First, it would select the features from the given training dataset, and get the best split to differentiate the observations based on the dependent variables. For example, in this project, the prediction variable is next VMT, and thus the features would be Month, Day, Hour, previous VHT, previous VHT, number of lane points, and percentage of observed cars. Then, under one feature, the binary decision tree would be applied to divide the space, which is a numerical process, and the impurity function and different split nodes help to select the optimal split. For the classification, the Gini impurity index function would be applied as an indication, which describes how mixed training data assigned to each node is [2]. As for the regression predictive modeling problem, the variance reduction would become the criterion that needs to be minimized so as to obtain the optimal split.

Thus, when the instances in the feature belong to the same class, then it would be set as a single node. Otherwise, the split would begin, the instance would compute the impurity once it is less than the threshold, the recursion would then stop and return the children node. Then, if the impurity is larger than the threshold, it would compute all features for this node, and obtain the minimum one as the optimal division, and it would go left or right depending on if it satisfies the conditions. Through this way, the main CART would be well-trained. Then, we would obtain the Huber loss and the corresponding gradient from this tree, and use the gradient and the original input to train another CART. After finishing the training, this CART could predict the "polishing gradient" by input the original data, and use the main CART to subtract the product of the learning rate and "polishing gradient". In this way, the main CART would fix some imperfect components. Then repeat the fixing process for n times, which depending on how many CART used in polishing, the main CART would be able to provide much better prediction, and the whole model is called GBDT. The test sample would follow the decision tree and finally stay in some children nodes, and the corresponding desired prediction value, VMT in this project, would be returned to the test samples as prediction.

## 5 Results

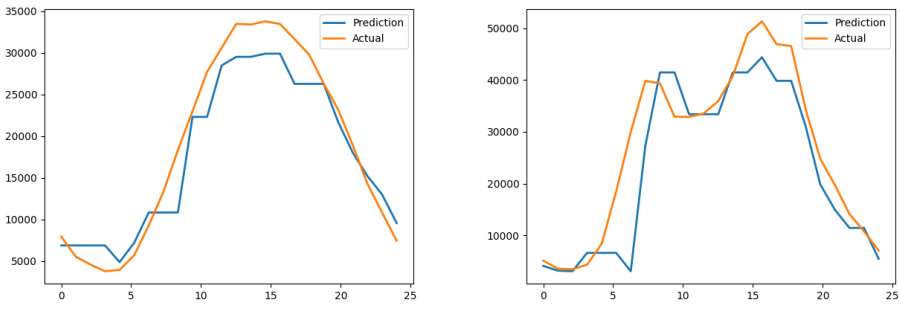
For the project, we trained a CART model and a GBDT model, and the result of each model would be shown below. Here, the general tendency of the CART model's prediction is quite similar as the




(a) (b)

(c)

Figure 4: Long term: (a) Relative error, (b) Histogram of error, (c) VMT prediction for whole July




(a) (b)

Figure 5: GBDT Prediction vs. Ground Truth on: a) date1, b) date2

actual data, and the relative error and the histogram of error are shown in Figure 4. Then, if we consider the prediction with error less than 20% as the proper prediction, then the accuracy of this prediction is 91.67%, which seems good but there still exists some large error. As for the GBDT model, the tendency and the shape would be more closer to the ground truth, especially for some peak values. Thus, we could conclude that the GBDT model would make more accurate prediction than the single CART model.

However, if we try to predict for a longer period, the performance of the model would be much worse. Figure 4 demonstrates the case for predicting the VMT for the whole July of 2021, and the corresponding results are as follows. Among 744 testing samples, only 442 had the acceptable error, so the accuracy would be only 59.41%, which seems unacceptable. In other words, it shows that our model is not suitable for prediction on long term traffic flow.

## 6 Conclusion

In conclusion, the conducted experiment proves that gradient boosting decision tree is indeed suitable for the short-term prediction of traffic flow on freeway which offers us help to adjust the strategy traffic control. However, we still need optimization on the algorithm to tune better model. The current model only predicts the overall tendency of the change of traffic flow through time, while having several sharp edges and horizontal segments. We believe that is caused by current pruning strategy. On the other hand, the variables using in this project are only concerned about the most relative features in ideal conditions. It is not quite enough since many parameters like the weathers, holidays that can definitely affect the driving conditions are no included. Besides, there are also some corner conditions, such as price of gasoline, are not considered. For further study,it is a wise choice to continuously explore the potential features of the data to come up with a best sets of features for training, concluding more potential parameters in the model, improving its accuracy. At last, neural network models that are also popular on short-term prediction, such as LSTM and GRU [4], should be concerned in ablation experiment to comprehensively evaluate the overall performance of prediction and expense of training a suitable model. Right now, due to insufficient time and late consideration on implementing other neural networks for comprehensive evaluation, the testing models perform poor which require more time on tuning to be ready for running the evaluation with GBDT.

## Contribution

The idea of this project, the machine learning based model for short term traffic prediction was determined by our group discussions. Among them, Chen is mainly responsible for programming, and the remaining two members, Peng and Wang are responsible for testing and writing reports.

## GitHub Repo

`https://github.com/AshtonYC/GBDT_TrafficFlowPrediction`

## References

[1] Loh, Wei-Yin. "Classification and regression trees." Wiley interdisciplinary reviews: data mining and knowledge discovery 1.1 (2011): 14-23.

[2] De'ath, Glenn, and Katharina E. Fabricius. "Classification and regression trees: a powerful yet simple technique for ecological data analysis." Ecology 81.11 (2000): 3178-3192. Dataset:

[3] Senyan Yang, Jianping Wu, Yiman Du, Yingqi He, Xu Chen, "Ensemble Learning for Short-Term Traffic Prediction Based on Gradient Boosting Machine", Journal of Sensors, vol. 2017, Article ID 7074143, 15 pages, 2017. https://doi.org/10.1155/2017/7074143

[4] Boukerche, Azzedine, and Jiahao Wang. "Machine Learning-based traffic prediction models for Intelligent Transportation Systems." Computer Networks 181 (2020): 107530.