Personalized Context-Aware Multi-Modal Transportation Recommendation

Meixin Zhu
University of Washington
Department of Transportation Engineering
meixin92@uw.edu

Jingyun Hu
University of Washington
Department of Transportation Engineering
jingyun@uw.edu

Abstract

This project proposes to find the most appropriate transport modes with awareness of user preferences (e.g., costs, times) and trip characteristics (e.g., purpose, distance). The work is based on real-life trips obtained from a map application. Several methods including gradient boosting tree, learning to rank, multinomial logit model, automated machine learning, random forest, and shallow neural network have been tried. For some method, feature selection and over-sampling techniques were also tried. The results show that the best performing method is a gradient boosting tree model with SMOTE over-sampling. This final report introduces the exploratory data analysis, method development, and results analysis along with discussion.

1 Introduction

Transport modes, such as walking, cycling, automobile, public transit, are means for traveling from an origin to a destination. Transportation mode recommendation refers to the effort of finding the most appropriate transport tools.

By **context-aware**, we aim to address the fact that the transport mode preferences change over various users and spatiotemporal contexts. For example, metros are more cost-effective than taxis for most urban commuters; economically disadvantaged people may prefer cycling and walking to others for local travel, if the transport options are inadequate (KDD Cup 2019). By **multi-modal**, we intend to address the limitation that existing transportation recommendation solutions only consider routes in one transportation mode. Imagine a scenario that the distance of the OD pair is relatively large, and the trip purpose is in no emergency. In this case, a cost-effective transportation recommendation that including multiple transport modes, e.g., taxi-bus, maybe more attractive, as shown in Figure 1.

In sum, context-aware multi-modal transportation refers to recommending a travel plan consists of various modes, such as walking, cycling, driving, public transit and ride sharing under various contexts. It can not only help users balance travel time and travel cost, but also contribute to reducing congestion, balancing traffic flow, and promoting the development of intelligent transportation system.

Specifically, given a user u, an origin-destination (OD) pair od, and the situational context, we want to recommend the most proper transport mode $m \in M$ for user u to travel between the OD pair od, considering user's preferences (e.g., costs, time) reveled in their historical trip data and trip characteristics (e.g., purpose, distance) (Liu et al., 2019). We will investigate this problem with large-scale navigation App data.

2 Related Work

There are a few studies related to transportation mode recommendations. Liu et al. (2019) proposed to recommend the most appropriate transport mode $m \in M$ for the user u to travel between the

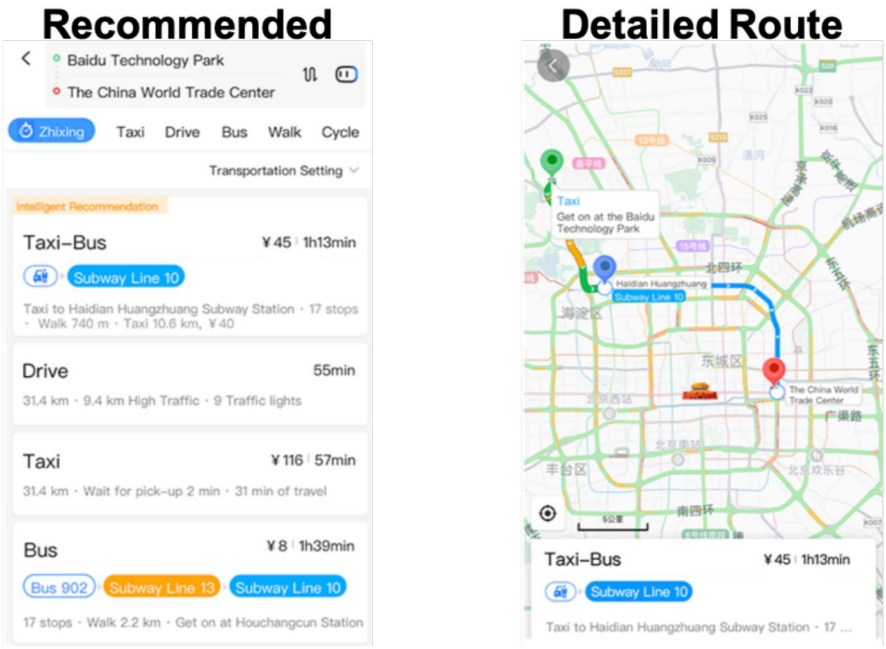

Figure 1: An example of user interfaces of context-aware multi-modal transportation recommendation service on Baidu Maps. The first recommended plan is 26.3% faster than the pure bus plan and 61.2% cheaper than the pure taxi plan. (KDD Cup 2019)

OD pair od. Specifically, they first extracted a multi-modal transportation graph from large-scale map query data to describe the concurency of users, OD pairs, and transport modes. Then, they developed embeddings for users, OD pairs, and transport modes based on networking embedding method. Finally, they exploited the learned representations for online multi-modal transportation recommendations. It is the very first work that formally define and solve the multi-mode transportation recommendation problem.

Cui et al. (2018) proposed to plan an optimal travel route between two geographical locations, based on the road networks and users' travel preferences. They defined users' travel behaviours from their historical Global Positioning System (GPS) trajectories and proposed two personalized travel route recommendation methods: collaborative travel route recommendation (CTRR) and an extended version of CTRR (CTRR+). The main drawback of this study is that they only considered single-mode transportation settings. This is one of the main challenge we are going to address in the current study.

Zhang et al. (2017) proposed a method to predict destinations of a user based on their starting location and time. The proposed method employs the Bayesian framework to model the distribution of a user's destination based on his/her travel histories. The main advantage is that they proposed a Bayesian framework to infer users' preferences based on historical data. However, they assume that the departure time and origin locations follow Gaussian distributions, and time and location are independent. This is might not be consistent with reality and thus is a drawback of the paper.

3 Data Description

Data in this project comes from KDD Cup 2019, which is provided by Baidu Map. The training set is the Baidu Map usage data from Oct. 1st to Nov. 30th, 2018. The testing set is data from Dec. 1st to Dec. 7th. Both the training and testing data was collected in Beijing, China.

The data consists of two main parts, user features data and a set of historical user behavior data. The user behavior data includes query records, display records and click records, as shown in Figure 2. A total of 500,000 query records were provided in the training data. Here is the detailed information in each data table:

sid	pid	req_time	o	d	plan_time	click_time	click_mode	plans
973273	100000	2018-10-24 20:58:45	116.37,40	116.39,39	2018-10-24 20:58:45	2018-10-24 20:59:2	10	distance price eta transport_mode 24205 1800 3807 10 17624 2306 3 17624 6400 2606 4 24332 500 4158 9 21859 700 4612 7 17374 500 5392 1
1203367	100000	2018-11-22 08:50:46	116.45,39	116.39,39	2018-11-22 08:50:46	2018-11-22 08:56:5	2	distance price eta transport_mode 11184 400 2629 2 13978 2596 2 9617 2674 3 9617 4900 2914 4
1695455	100000	2018-10-23 17:36:19	116.36,40	116.37,40	2018-10-23 17:36:19			distance price eta transport_mode 1735 524 6 2046 428 3 2046 1300 548 4 1445 1255 5 1892 200 1420 1 2491 200 1694 1

Figure 2: A preview of the user behavior data.

Query Records. A query record represents one route search from a user. Each query record consists of a session ID, a profile ID, a time stamp, and the coordinates of the origin-destination points.

Display Records. A display record is the list of routes generated by Baidu Maps shown to the user. Each display record consists of a session ID, a time stamp and a list of route plans. Each display plan consists of the transport mode (1 to 11), the estimated route distance (m), the estimated time of arrival (ETA) (s), the estimated price (RMB cent) and the display rank in the display list. There are 11 transport modes in total. A transport mode could be unimodal (e.g., drive, bus, cycle) or multi-modal (e.g., taxi-bus, cycle-bus). For privacy issue, the exact meaning of the 11 transportation modes are not provided.

Click Records. A click record is the specific transportation plan clicked by the user out of the whole list. A record contains a session ID, a time stamp, and the first clicked transport mode in the display list by the user.

User Features. User profile features reflect individual preference on transport modes. The user of each session is associated with a set of user attributes via a profile ID. Each profile record consists of a profile ID, a set of one hot encoded user profile dimensions. For the privacy issue, the real-world meaning of each attribute is not given. Also, users with same attributes are merged, sharing the same user profile ID. For example, with gender and age attribute considered, two males of age 35 are identified as the same user in the dataset.

4 Exploratory Data Analysis

This section provides some initial findings and summary statistics for the dataset.

In this study, the exact meaning of the 1 to 11 transportation modes is not provided, and the only features for each transportation mode are distance, time, and price. For transportation mode recommendation, these provided features are not enough because many factors that are closely related to people's mode choice behavior are ignored, such as riding comfort and time reliability. Therefore, we tried to infer the meaning of the provided transportation modes based on traveling price and speed derived from the raw data.

Figure 3 presents the mean price and mean speed information for each transportation mode. Based on common sense and Baidu Map Apps, we can making the following inferences:

- Mode 3 is driving by one's own vehicle because the price is zero, and the speed is the highest. Mode 4 is by taxi because of its high price and similar speed with driving.
- Mode 5 and 6 are walking and biking respectively because they are free with low speeds.
- Mode 1 and 2 should be transit bus and metro respectively because they are cheap, popular and with lower speeds compared to driving.
- Similarly, we can identify that mode 7 is metro-bus; mode 8 is bus-taxi, mode 9 is metro-bike, mode 10 is metro-taxi, and mode 11 is metro-bus-bike.

These inferences will be valuable for upcoming analysis because they provide additional information about our prediction targets.

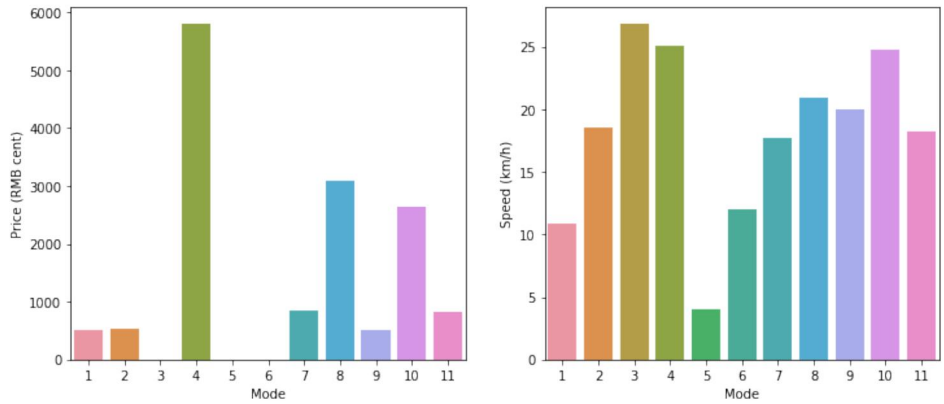

Figure 3: Mean price and traveling speed for each transportation mode.

Figure 4 shows the frequency people's mode choices in ground truth data (i.e., users' click mode). We can see that Metro, Metro+bus, and Bus are the most frequent modes people clicked, indicating that public transportation like transit and metro serves the majority of travelers.

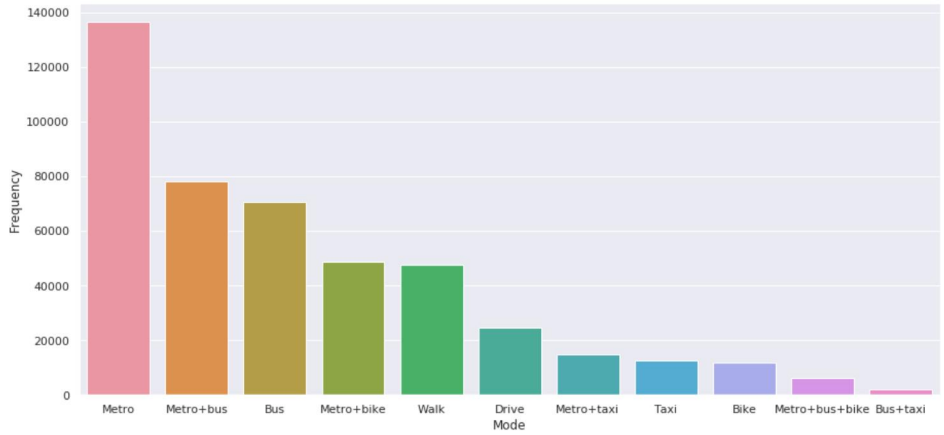

Figure 4: Frequency of mode choice in ground truth data.

Based on the data, we can explore some activity features of the map users. By drawing a heatmap of od distribution during morning and evening peak hours, as shown in Figure 5, several interesting facts were found. There are some specific trip attraction points in the morning which are identified as the Beijing Airport, the Great Wall and the Xiangshan Mountain. During the evening peak hours, more users are traveling from the airport instead. Also, the suburban area have more origin points in the morning and more destination points in the evening.

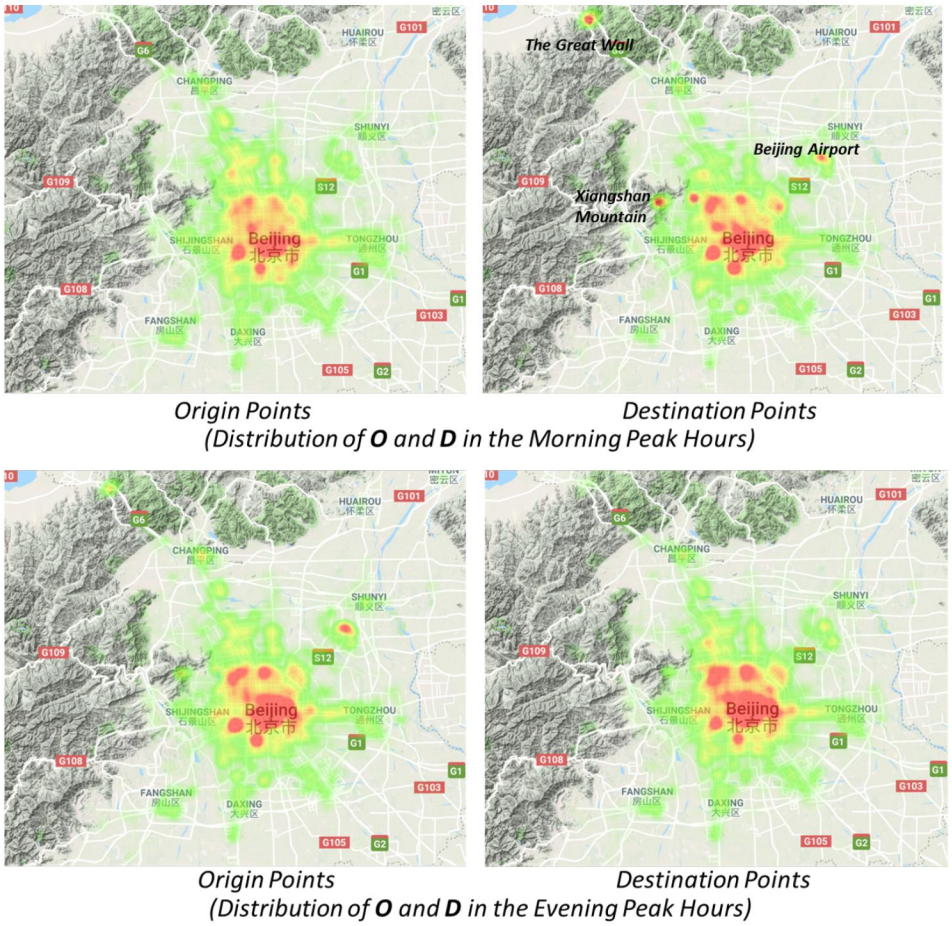

Figure 5: Heatmap of Origin and Destination points during Peak Hours.

5 Method

We built several methods for this problem (see Table 1), including gradient boosting tree methods, learning to rank, multinomial logit model, automated machine learning, random forest, and shallow neural network. Feature selection and over-sampling techniques were also tried.

Our main method is a gradient boosting tree model based on (**?**), treating the recommendation as a multi-class classification problem. is a gradient boosting framework that uses tree based learning algorithms. It is designed to be distributed and efficient with faster training speed and higher efficiency.

Basically, we first did feature engineering to generate features for the query records, treating them as X. Then we take users' click mode as y, and do multi-class classification training and prediction. The final model includes 304 features and can be summarized as follows:

- **Request time features:** which hour and which weekday.
- **Location features:** longitudes and latitudes of the OD pairs.
- **Transport mode features:** distance, price, time, price*time and speed for each candidate mode; descriptive statistics of distance, price, time, speed for all the modes in each plan, like the maximum, minimum, mean, and standard deviation of distances; which mode ranked first, which mode had longest and shortest travel distances respectively; which mode take the longest and shortest time respectively, which mode had the lowest and highest price respectively.

- **Metro and bus station features:** distances to the 5 nearest bus/metro stations for origin and destination locations; number of bus/metro stations within 1500 meters' range.
- **Point of interest (POI) features:** number of different nearby POIs for each origin and destination location. Used geohash to represent a location using a short alphanumeric string so that the 2 locations with the same string are considered as close to each. A total of 33 POI categories, including car service area, transportation ports (airports, railway stations), tourist area, shopping mall, etc., were used.
- **Location frequency features:** use geohash to discretize the area and count the visit frequency for each origin and destination area.
- **Important location features:** the distance between origin/destination location and the important trip attraction locations (Beijing Airport, the Great Wall, Xiangshan Mountain, etc.) identified through visit heat maps described in Section 3.
- **Profile features:** binary user profile features.

The top 10 important features in the final LightGBM model are shown in Table 7, with their importance scores.

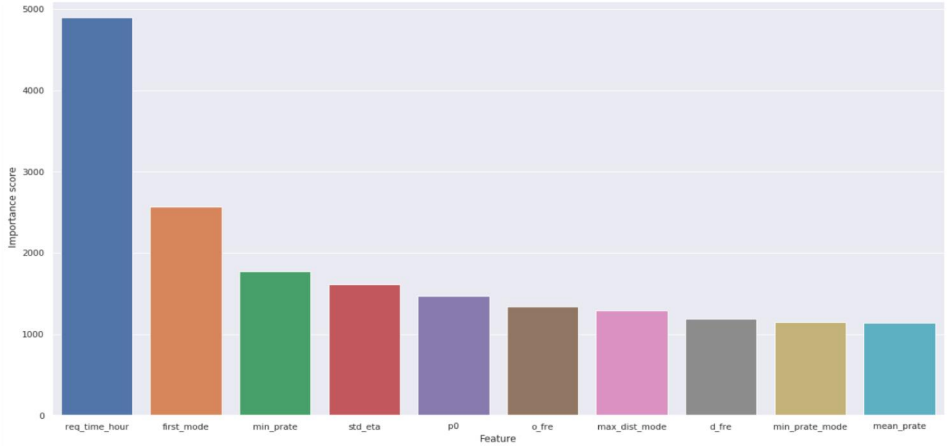

Figure 6: Top 10 important features.

One important characteristic of this problem is the unbalanced data distribution of the clicked transport modes. With a initial analysis, we found that the model's performance on each mode is strongly correlated with the sample size of each mode: the frequently clicked modes have a higher accuracy than the less frequent ones. To address this issue, we utilized a over-sampling technique called Synthetic Minority Oversampling Technique (SMOTE) (Chawla et al., 2002). Basically, the method uses K-nearest neighbors to generate new samples for the minority classes. Compared to directly copy existing data samples, denoted as random over-sampling, which degrades the model's performance, the SMOTE method can improve the model's performance as lot, as shown in Table 1.

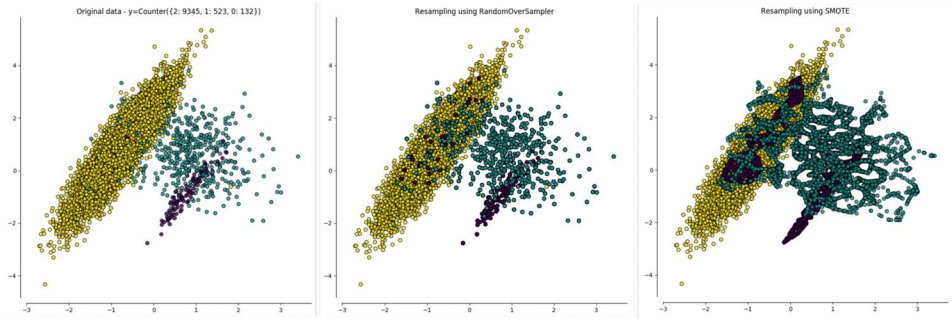

Figure 7: Top 10 important features.

The detailed hyperparameters with respect to the graident boosting tree methods are listed as follows : num of leaves = 40, max depth=8, learning rate = 0.1, subsample rate = 0.8, feature selection ratio = 0.8, min child samples = 60.

6 Results

In this study, weighted F1 is used as the evaluation score. The F1 score for each class is defined as

$$F_{1,\text{Class}_i} = \frac{2\text{precision} * \text{recall}}{\text{precision} + \text{recall}} \tag{1}$$

where precision and recall are calculated by counting the total true positives, false negatives and false positives globally. The weighted F1 is calculated by considering the weight of each class,

$$F_{1,\text{weighted}} = w_1 F_{1,\text{ class }_1} + w_2 F_{1,\text{ Class }_2} + \cdots + w_k F_{1,\text{ Class }_k} \tag{2}$$

the weight is calculated by the ratio of true instances for each class.

We used data from Oct. 1st to Nov. 23th, 2018 as training data, and data from Nov. 24th to Nov. 30, 2018 as validation data. We then used the model that performed best on the validation data to predict mode choices on test data (from Dec. 1st to Dec. 7th, 2018).

The performances of different methods are shown in Table 1. The lightgbm moethod with smote over-sampling performed the best.

Table 1: Performance of Different Models on testing data.

Model	Weighted F1 score
Lightgbm with smote oversample	**0.6951**
Lightgbm	0.6920
Lightgbm with backward feature selection	0.6912
Lightgbm with pid	0.6905
Lightgbm with random oversample	0.6884
Lightgbm learning to rank	0.6883
Xgboost (Chen and Guestrin, 2016)	0.6877
Random forest (Breiman, 2001)	0.6851
Auto ML (Guyon et al., 2015)	0.6836
Catboost (Dorogush et al., 2017)	0.6835
Xgboost learning to rank	0.6661
train with single pid (if not have enough data, grouped with others)	0.6645
Multinomial logit model	0.4971
Shallow neural network (6 layers)	0.0178

The performance of our main method (Lightgbm with smote over-sampling) on different transportation modes is shown in Table 2. The metro mode has the highest performance maybe because it has more data samples. We can see that prediction on drive, taxi, and bike should be improved in future because of their extremely low F1 scores. With this this main method, our online testing F1 score is 0.6951, ranking 132/1702 on the final leaderboard, as shown in Figure 8.

Table 2: Lightgbm model performance for each transportation mode on validation data.

Mode	Sample ratio	F1 score	Precision	Recall
No click	0.0874	0.3552	**0.9453**	0.2187
Bus	0.1446	0.6804	0.6372	0.7299
Metro	**0.3133**	**0.9019**	0.8543	**0.9551**
Drive	0.0446	0.1424	0.3997	0.0867
Taxi	0.0245	0.0829	0.1836	0.0535
Walk	0.0976	0.8452	0.7859	0.9143
Bike	0.0199	0.2187	0.2529	0.1927
Metro+bus	0.1779	0.7883	0.7112	0.8840
Bus+taxi	0.0046	0.3288	0.2568	0.4567
Metro+bike	0.0499	0.5148	0.5803	0.4626
Metro+taxi	0.0285	0.5424	0.4643	0.6521
Metro+bus+bike	0.0072	0.4187	0.3731	0.4771

Validation F1 score for all modes: **0.693168348**

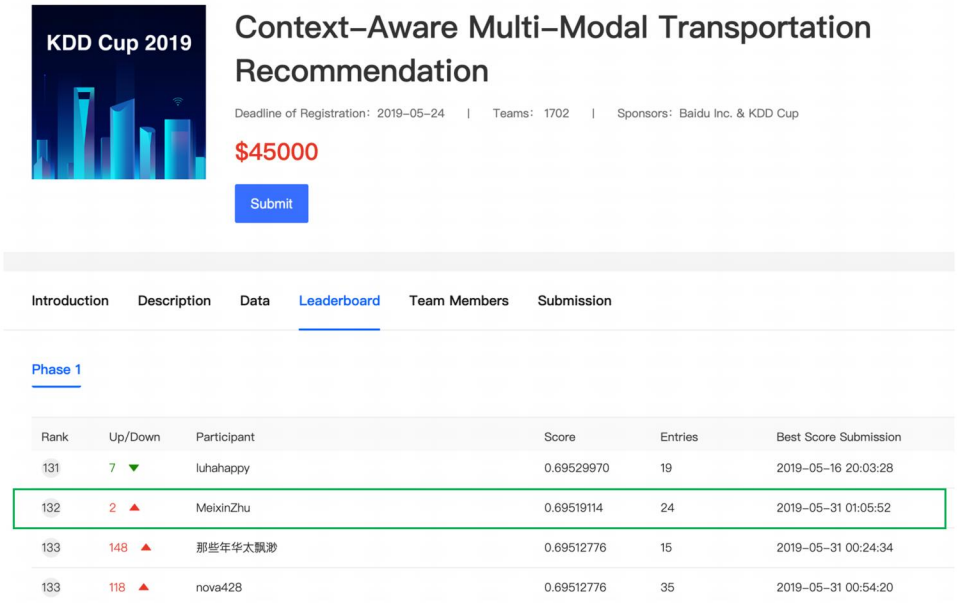

Figure 8: Online ranking for KDD Cup 2019.

7 Summary and Discussion

This project aims to find the most appropriate transport modes with awareness of user preferences (e.g., costs, times) and trip characteristics (e.g., purpose, distance). Several methods have been tried, and the best performing method is a gradient boosting tree model with SMOTE over-sampling. Below are some further discussion points:

- Currently, the highest score in the leaderboard is 0.7043, still far from perfect.
- Not sure the ceiling of the F1 score for this problem. Maybe human's behavior is too hard to predict, and the F1 score is impossible to be larger than 80%.
- Oversampling gives a stable boosting for the model performance.
- Individual heterogeneity does exist.
- Things tried but not having significant effects (or even making the performance worse):
  - Do PCA before learning

  - Expand the plan list and do binary classification
  - Adding class weights for unbalanced data
  - Incorporate historical mode choice probability as features
  - Scale the data
  - Normalize the data
  - Scale the time, price, distance to 0 1 within the candidate plans
  - SVM (too slow)
  - Train with individual's data. Turned out that some pids are just too hard to predict
  - Adding weather as features
- More combination of the variables should be tried, like interaction terms, transformation of the variables (e.g. $log(X)$, X^2 instead of X).

For future work, our main efforts will focus on using embedding learning method for graph. We will try to better represent transportation modes, users, and OD pairs, so that classification models can have better performances.

8 Contributions of Team Members

Meixin Zhu: Data preprocess , coming up with and coding up the algorithm, writing up the report.

Jingyun Hu: Data exploration, optimizing the algorithm, writing up the report.

Members contribute equally in this project.

References

Liu, H., T. Li, R. Hu, Y. Fu, J. Gu, and H. Xiong, Joint Representation Learning for Multi-Modal Transportation Recommendation. *AAAI, to appear*, 2019.

Cui, G., J. Luo, and X. Wang, Personalized travel route recommendation using collaborative filtering based on GPS trajectories. *International journal of digital earth*, Vol. 11, No. 3, 2018, pp. 284–307.

Zhang, L., T. Hu, Y. Min, G. Wu, J. Zhang, P. Feng, P. Gong, and J. Ye, A taxi order dispatch model based on combinatorial optimization. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ACM, 2017, pp. 2151–2159.

Chawla, N. V., K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, SMOTE: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, Vol. 16, 2002, pp. 321–357.

Chen, T. and C. Guestrin, Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, ACM, 2016, pp. 785–794.

Breiman, L., Random forests. *Machine learning*, Vol. 45, No. 1, 2001, pp. 5–32.

Guyon, I., K. Bennett, G. Cawley, H. J. Escalante, S. Escalera, T. K. Ho, N. Macià, B. Ray, M. Saeed, A. Statnikov, and E. Viegas, AutoML Challenge 2015: Design and first results. In *Proc. of AutoML 2015@ICML*, 2015.

Dorogush, A. V., A. Gulin, G. Gusev, N. Kazeev, L. O. Prokhorenkova, and A. Vorobev, Fighting biases with dynamic boosting. *CoRR*, Vol. abs/1706.09516, 2017.