Hydra: A Personalized and Context-Aware Multi-Modal Transportation Recommendation System

Hao Liu[1], Yongxin Tong[2], Panpan Zhang[1], Xinjiang Lu[1], Jianguo Duan[1], Hui Xiong[1,3*]
[1]The Business Intelligence Lab, Baidu Research,
National Engineering Laboratory of Deep Learning Technology and Application, Beijing, China,
[2]SKLSDE Lab, Beihang University, Beijing, China, [3]Rutgers University
[1]{liuhao30, zhangpanpan04, luxinjiang, duanjianguo}@baidu.com,
[2]yxtong@buaa.edu.cn, [3]xionghui@gmail.com

ABSTRACT

Transportation recommendation is one important map service in navigation applications. Previous transportation recommendation solutions fail to deliver satisfactory user experience because their recommendations only consider routes in one transportation mode (uni-modal, *e.g.*, taxi, bus, cycle) and largely overlook situational context. In this work, we propose Hydra, a recommendation system that offers multi-modal transportation planning and is adaptive to various situational context (*e.g.*, nearby point-of-interest (POI) distribution and weather). We leverage the availability of existing routing engines and big urban data, and design a novel two-level framework that integrates uni-modal and multi-modal (*e.g.*, taxi-bus, bus-cycle) routes as well as heterogeneous urban data for intelligent multi-modal transportation recommendation. In addition to urban context features constructed from multi-source urban data, we learn the latent representations of users, origin-destination (OD) pairs and transportation modes based on user implicit feedbacks, which captures the collaborative transportation mode preferences of users and OD pairs. A gradient boosting tree based model is then introduced to recommend the proper route among various uni-modal and multi-modal transportation routes. We also optimize the framework to support real-time, large-scale route query and recommendation. We deploy Hydra on Baidu Maps, one of the world's largest map services. Real-world urban-scale experiments demonstrate the effectiveness and efficiency of our proposed system. Since its deployment in August 2018, Hydra has answered over a hundred million route recommendation queries made by over ten million distinct users with 82.8% relative improvement of user click ratio.

KEYWORDS

Transportation recommendation; context-aware; personalized; feature engineering; deployment

ACM Reference Format:
Hao Liu, Yongxin Tong, Panpan Zhang, Xinjiang Lu, Jianguo Duan, and Hui Xiong. 2019. Hydra: A Personalized and Context-Aware Multi-Modal Transportation Recommendation System. In *The 25th ACM SIGKDD Conference on Knowledge Discovery & Data Mining (KDD'19), August 4–8, 2019, Anchorage, AK, USA.* ACM, NY, NY, USA, 9 pages. https://doi.org/10.1145/3292500.3330660

1 INTRODUCTION

Transportation recommendation is a core component in various map services and has deeply penetrated into the everyday life of citizens. Transportation recommendation refers to a set of routes recommended to users given the specific OD pair input by users. Online map services such as Baidu Maps answer over a hundred million transportation recommendation queries made by over ten million distinct users in China per day.

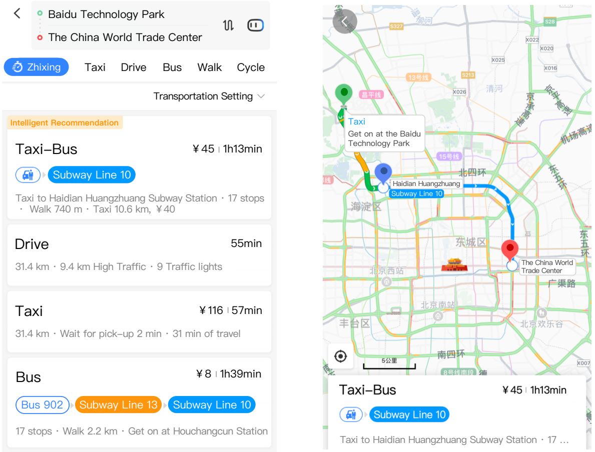
Despite its popularity and frequent usage, existing transportation recommendation solutions still fail to deliver satisfactory user experience. According to Baidu Maps, over 15% of the users tend to request transportation recommendations on different uni-modal routing engines (e.g., taxi and bus), indicating the requirement of inter-modal transportation comparison. Furthermore, 89.1% routing queries from users are answered with feasible transportation recommendations but over 58.5% of the transportation recommendation list has no user clicks (see Table 1), indicating none of the recommended transportation plans are satisfactory.

The above observations indicate two limitations of current transportation recommendation solutions. *(i) Ignorance of situational context.* For instance, when a big concert lets out, it is difficult to call a taxi. A better solution may consider supplement of multiple alternative transportation modes (as illustrated in Figure 1) and recommend the most efficient one. *(ii) Uni-modal transportation recommendation.* For example, imagine the following 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 illustrated in Figure 1(b)). Hence, the transportation recommendation should adapt to the situational context *e.g.*, whether there is a concert, and provides more flexible recommendations, *e.g.*, combining buses and taxis.

To address these limitations, we propose Hydra, a personalized and context-aware multi-modal transportation recommendation system. Inspired by the availability of existing routing engines and big urban data, we design a novel framework that integrates route plans in different transportation modes (including both uni-modal and multi-modal transportation plans) and heterogeneous

[*] Corresponding author.

KDD '19, August 4–8, 2019, Anchorage, AK, USA
© 2019 Association for Computing Machinery.
ACM ISBN 978-1-4503-6201-6/19/08...$15.00
https://doi.org/10.1145/3292500.3330660



Baidu Technology Park
The China World Trade Center

Taxi–Bus ¥45 | 1h13min
Subway Line 10
Taxi to Haidian Huangzhuang Subway Station · 17 stops · Walk 740 m · Taxi 10.6 km, ¥40

Drive 55min
31.4 km · 9.4 km High Traffic · 9 Traffic lights

Taxi ¥116 | 57min
31.4 km · Wait for pick–up 2 min · 31 min of travel

Bus ¥8 | 1h39min
Bus 902 Subway Line 13 Subway Line 10
17 stops · Walk 2.2 km · Get on at Houchangcun Station

(a) Recommendation list.

Taxi–Bus ¥45 | 1h13min
Subway Line 10
Taxi to Haidian Huangzhuang Subway Station · 17 …

(b) Details of the first route.

Figure 1: An example of user interfaces of Hydra **on Baidu Maps. The left figure shows the list of plans in various transportation modes ordered by our recommendation model. The right figure shows the details of the top-**1 **recommendation, which is a multi-modal transportation plan (*i.e.*, first take taxi and then bus). The first recommended plan is** 26.3% **faster than the pure bus plan and** 61.2% **cheaper than the pure taxi plan.**

urban data. Hydra not only extracts features from multi-source urban data, but also learns the latent representations of users, OD pairs and transportation modes based on user clicks to capture the collaborative transportation mode preferences of users and OD pairs. It then applies a gradient boosting tree based model to recommend the proper transportation plan among various uni-modal and multi-modal transportation routes.

In web-scale recommendation, the service scalability and online recommendation latency is also curial for user experience [22]. To address the service efficiency concern, we build a distributed offline data pipeline as well as an RPC based online web service framework. Besides, we propose a dedicated region index structure in online feature processing to reduce the online recommendation latency.

Our main contributions can be summarized as follows.

- We propose Hydra, a multi-modal transportation recommendation system. To the best of our knowledge, this is the first product level intelligent routing engine that integrates various transportation modes in a unified service.
- We design a novel recommendation model that is adaptive to the situational context. We extract a rich set of features from multi-source urban data to sense the context variation and adopt a graph embedding based algorithm to capture the transportation preferences of users and OD pairs.
- We propose a series of optimization techniques to improve the time efficiency of the recommendation system and discuss several deployment issues in a hundred million user level online map service.
- Extensive real-world urban-scale experiments on real datasets show that our proposed framework outperforms six baseline algorithms in four metrics. The online recommendation service achieves less than 250ms latency in average and scale well in the production environment.

Table 1: Statistics of datasets.

Data description		Beijing	Shanghai
User behavior data	# of queries	5,956,596	5,628,921
	# of displays	5,308,127	4,993,350
	# of clicks	2,205,091	1,980,870
Geographical data	# of POIs	900,669	1,061,399
	# of road segments	812,195	768,336
	# of bus stations	44,830	45,052
Meteorological data	# of weather records	34,944	32,760
User profile data	# of distinct users	1,199,399	1,217,140

In the rest of this paper, we first describe the details of the real-world large datasets used in our study in Section 2, and elaborate on the detailed methodologies of our transportation recommendation system in Section 3. We next introduce our deployment details and efficiency optimization techniques in Section 4. Evaluations on two large-scale industrial datasets are presented in Section 5. Finally we review related work in Section 6 and conclude in Section 7.

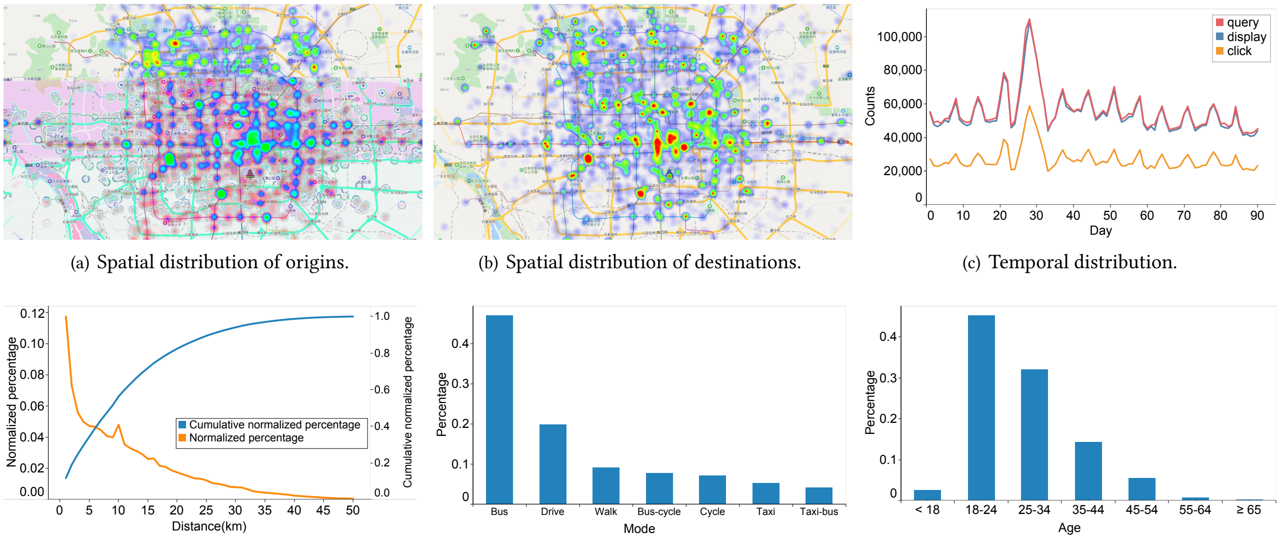
2 DATA DESCRIPTION AND ANALYSIS

This section introduces the datasets that will be used in the following sections, with a preliminary data analysis. The datasets include user behavior data, geographical data, meteorological data and user profile data collected from Beijing and Shanghai, two metropolises in China. Table 1 summarizes the statistics of the datasets.

2.1 User Behavior Data

User behavior data captures the user interactions with navigation applications. Our user behavior data are collected from Baidu Maps, a large-scale navigation app, from September 2018 to November 2018. According to a user interaction loop, the user behavior data can be further categorized into *query records*, *display records* and *click records*. In short, a query record represents one route search from a user on Baidu Maps; a display record is the routes recommended by Baidu Maps shown to the user; and a click record indicates the user feedback of different recommendations (*i.e.*, a user may click on specific routes displayed to him/her for details, as in Figure 1). Please refer Appendix A for detailed data description.

We briefly explain the distributions of our user behavior dataset in Beijing below (see Figure 2(a)-Figure 2(e)). Note that similar observations held in Shanghai, which we omit due to the page limit. Figure 2(a) and Figure 2(b) depict the spatial distributions of origins and destinations in the query records. Most origins and destinations are within the 6th ring road, *i.e.*, the central area of Beijing. The distribution of destinations is more concentrated than that of origins, indicating that most queries are about specific POIs such as transport stations and city landmarks. The spatial distribution patterns of origins and destinations motivate us to use geographical data to capture the spatial dependency for transportation route recommendation. Figure 2(c) plots the temporal distributions of query, display and click records (*i.e.*, numbers per day). The temporal distributions exhibit strong periodicity, where peaks often correspond to weekends and holidays. For example, the peaks on the 22nd and 31st days correspond to the mid-autumn festival and



(a) Spatial distribution of origins.

(b) Spatial distribution of destinations.

(c) Temporal distribution.

(d) Distance distribution.

(e) Transport mode distribution.

(f) User age distribution.

Figure 2: Distributions of the Beijing dataset: (a) the spatial distribution of query origins; (b) the spatial distribution of query destinations; (c) the temporal distribution of user query behavior in three successive months; (d) the distribution of travel distances; (e) the distribution of clicks on transport modes; (f) the distribution of user ages.

the National day, two public holidays in China. Figure 2(d) shows the distribution of trip distance from the queries. Here the trip distance is measured by the spherical distance on earth [10]. Over 60% trips are within 10 Kms and 80% trips are within 20 Kms. This indicates short-distance and mid-distance trips are the major query demand on online navigation applications. Figure 2(e) shows the distribution of clicks on different recommended routes. Above 54.64% clicks involve buses (*i.e.*, bus and bus-bicycle) and 25.12% clicks are drive or taxi, indicating public and car-based transportation are more preferable.

2.2 Geographical Data

Intuitively, geographical characteristics of origins and destinations partially reflect the situational context, and thus affect user preferences on transportation modes. Accordingly, we use a large-scale geographical dataset collected from *(i)* professional surveyors employed by Baidu Maps *(ii)* the crowdsourcing platform in Baidu, which include *POI data*, *road network data* and *transportation station data* in Beijing and Shanghai. All data are updated daily. We present a detailed data description in Appendix A.

2.3 Meteorological Data

Meteorological data tend to reflect the temporal dynamics of the situational context when planning trips, and thus may also affect the user preference on transportation modes. For example, the demand for taxis may be higher in the case of snow, rain and severe air pollution. We collect the meteorological data from an online meteorology website of the Chinese government over three months from September 1st to November 30th. Each record of meteorological data consists of an administrative district, a time stamp, the weather, the temperature, the wind strength, the wind direction and the Air Quality Index (AQI). The weather is categorized as sunny, cloudy, rainy and overcast. The AQI is an integer of the air pollution level.

2.4 User Profile Data

User profile attributes reflect individual preference on transportation modes. For instance, subways are more cost-effective than taxis for most urban commuters, and driving is likely to be the first choice for car owners. We collect user profile attributes from multiple Baidu applications including Baidu search, Baidu App and Baidu Maps. The Beijing dataset contains 1, 199, 399 distinct user records and the Shanghai dataset contains 1, 217, 140 distinct user records. Each record consists of a user’s demographic attributes including the age, the gender, and social attributes such as the industry, the educational level, and whether the user is a car owner. All user profile records are anonymized and cannot be associated with sensitive personal information such as names and phone numbers. Figure 2(f) plots the age distribution of Beijing dataset. Most Baidu Maps users are between 18 and 54 years old.

3 HYDRA FRAMEWORK

This section presents the framework of Hydra in detail.

3.1 Overview

Figure 3 shows an overview of Hydra. It consists of four major components, *Route generation*, *Feature construction*, *Transport mode preference representation* and *Transportation recommendation*. The *Route generation* module leverages existing uni-modal routing engines to generate feasible routes in different transport modes. Thereafter, the *Feature construction* module extracts features from various urban datasets. Meanwhile, the *Transport mode preference representation* module captures high-order user (resp. OD pair)

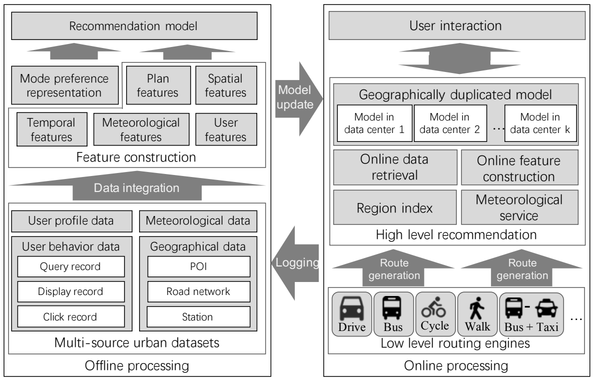


Figure 3: Hydra **Overview.**

transport mode preference representation through a graph embedding method. Finally, the *Transportation recommendation* module integrates handcrafted features and embedding features to make recommendation. In this paper, we consider seven transport modes $\{drive, taxi, bus, cycle, walk, taxi\text{-}bus, bus\text{-}cycle\}$. In particular, the first five modes are uni-modal transport modes whereas *taxi-bus* and *bus-cycle* are multi-modal transport modes. According to log analysis, *taxi-bus* and *bus-cycle* are top multi-modal travel demands and are already well supported in Baidu Maps. Note that we treat each uni-modal and multi-modal transport mode as distinct transport modes, which makes our model extendable for other potential transport modes.

3.2 Route Generation

We adopt existing low level routing engines to generate feasible routes for each transport mode. In online processing, the route is searched in real-time in parallel. Specifically, when a query is received, a station binding process is first applied to bind origin and destination locations to validate start and end points. For example, the location is bound to road segments for drive and taxi, and to transport stations for bus. After that, a bidirectional shortest-path search [12] is applied on each transportation network. For each uni-modal transport mode, a contraction hierarchy (CH) [11] is pre-constructed on the transportation network to reduce search latency. A set of valid routes is generated by various criteria, *e.g.*, fastest, distance shortest and least transfer. For multi-modal transportation, the search strategy is slightly different. We build a multi-modal transportation network [6] and restrict the number of modal-transfer to guarantee the utility [5] of searched routes. Finally, an internal rule based ranking model is applied in each transport mode to filter out routes with high segment overlap and decide the order of routes. For ease of cross-mode comparison, only one route of each transport mode will appear in the final display. In offline processing, the route is directly retrieved from user behavior data. In the production environment, a query understanding component will be invoked before route generation to bind fuzzy search keywords with concrete POIs. We omit further discussions since they are out of the scope of transportation recommendation.

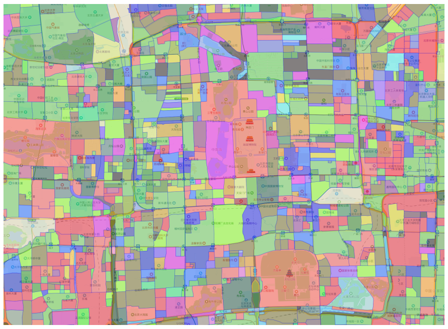


Figure 4: An illustrative example of region partition base on road network connectivity in Beijing. Different colors indicate different region functionality derived by *TF-IDF* based on the POI distribution.

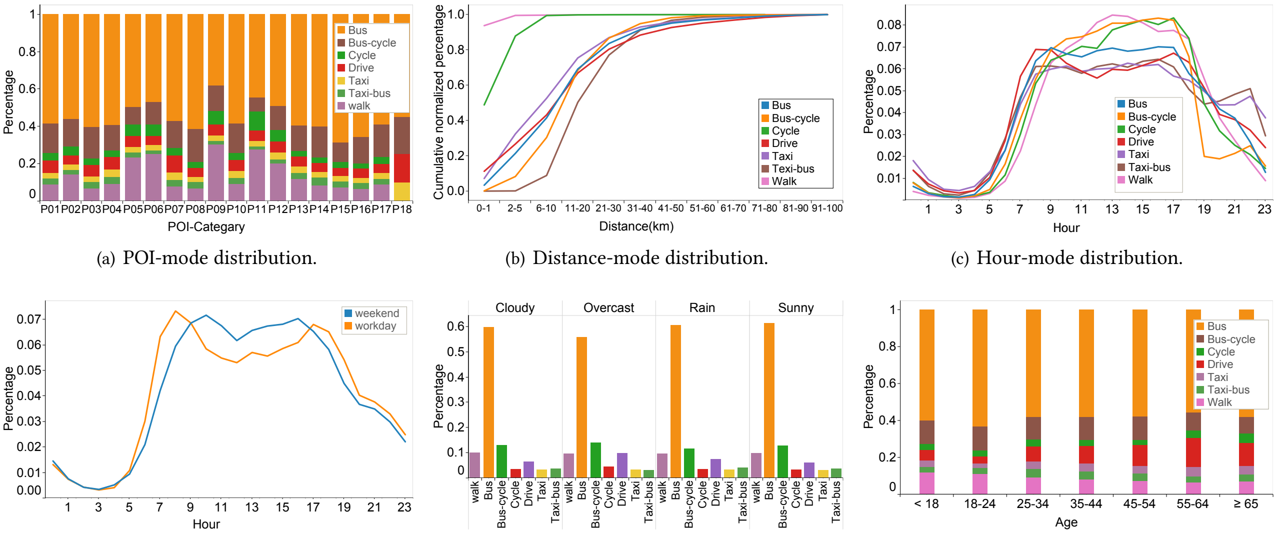
3.3 Feature Construction

We introduce the process of constructing, transforming and augmenting feature vectors below. Appendix B lists features we construct based on each dataset with a detailed description.

3.3.1 Plan Features. Cost of a plan such as *Price* and *ETA* are part of considerations for user preferences. For each plan, we extract *Road network distance, Route distance, ETA, Price, Transfer count, Transfer model count* from display records. The *Road network distance* is the real travel distance on the road network. For walking and cycling, *Price* is set to zero.

3.3.2 Spatial Features. We first extract *District* and *POI category* features of the origins and destinations. As shown in Figure 5(a), the transportation mode choices of different destination POI categories vary. For example, the demand for buses to *Sports* and *Tourist Attraction* POIs is higher than average. In contrast, the demand for buses to *Beauty*, *Life Service* and *Food* POIs is lower than average. Then we calculate the *Spherical distance* of OD pairs. Figure 5(b) shows the relation between trip distance and the percentage of different transport modes. We observe a strong correlation between Spherical distance and transport mode choice. Walk and cycle are the major choices for trips shorter than 5 Km whereas bus and drive are the major choices for trips longer than 10 Km. The peak of demand for taxi appears when the trip distance is near 5 Km. Since the road connectivity and transport stations in a region are fixed, the transport availability of adjacent OD pairs is similar. To incorporate such regional dependency, we partitioned the city into a set of non-overlapping regions through the road network [34]. Figure 4 gives an illustrative example of partitioned regions of Beijing. For each origin region, we further compute the POI count of each POI category as *Regional POI distribution*, transport facility count (*i.e.*, road segment, road intersection, bus station and bus line) as *Regional transport facility distribution* and transport mode click count as *Regional historical mode distribution.* We also extract similar features for destination regions and OD region pairs.

3.3.3 Temporal Features. We exploit *Hour, Minute, Day of week, Day of month* and *Workday* as the temporal features. As shown in Figure 5(c), distributions of transportation mode choices differ in different time periods. The demand for walk and cycle is mainly in daytime whereas the demand for taxi and taxi-bus is still high at



(a) POI-mode distribution.

(b) Distance-mode distribution.

(c) Hour-mode distribution.

(d) Workday & weekend-mode distribution (Drive).

(e) Weather-mode distribution.

(f) Age-mode distribution.

Figure 5: Feature distributions of the Beijing dataset.

night. As illustrated in Figure 5(d), the transport mode preferences during different time periods on weekdays and weekends also differ. For drive, there are two peak hours in a day. However, the peak on weekday mornings is earlier than that on weekend mornings and the peak on weekday evenings is later. Conversely, peak hours at weekends are closer and the demand is more evenly distributed in the daytime.

3.3.4 Meteorological Features. We adopt *Weather, Temperature, AQI, Wind speed* and *Wind direction* as the meteorology features. Figure 5(e) depicts the correlation between weather and transport mode preference distributions. The demand for drive is higher on overcast and rainy days whereas the demand for bus on overcast days is lower.

3.3.5 User Features. We construct user features based on users' *Demographic attribute, Social attribute* and *User historical mode distribution*, as shown in Table 5. Figure 5(f) depicts the correlation between the age of users and the transport mode choices. We observe that older people have higher demand for drive and taxi, whereas younger people prefer walk and bus more.

3.4 Transport Mode Preference Representation

Transport mode preference representation aims to learn high order collaborative relationship among users, OD pairs, and transport modes. The intuition is, users travelling similar OD pairs via similar transport modes have similar transport mode preference. Inspired by the recent success of embedding methods [13, 18] on preserving local network structures, we construct a heterogeneous graph $G = (\mathcal{V}, \mathcal{E})$ of user nodes $\mathcal{U}$, OD pair nodes $\mathcal{OD}$ and transport mode nodes $\mathcal{M}$ based on the user behavior data (Figure 6). The target is to project each node $v \in G$ into a low dimensional vector in the latent space, each of which reflects the neighborhood relationship (a.k.a. the second-order proximity) in G. We analogize random walks with the constructed click events, where a *click event* is defined as a user u clicked on a route in transport mode m over a specific OD pair od. We adopt Trans2vec [15] and skip-gram [17] on G. Specifically, given a click event, the latent vectors of $v^u \in \mathcal{U}$, $v^{od} \in \mathcal{OD}$ and $v^m \in \mathcal{M}$, denoted as $\mathbf{u}^u$, $\mathbf{u}^{od}$, and $\mathbf{u}^m$, are learned by maximizing the following conditional log probability:

$$O_t = \sum_{t \in T} \sum_{v_i \in V^t} \sum_{n_j^t \in N_t(v_i)} \log p(n_j^t | v_i), \qquad (1)$$

where $T = \{u, od, m\}$ is the type of nodes in G, and $n_j^t \in N^t(v_i)$ is the type aware context node of v_i ever co-occurred in a click event. That is, only heterogeneous neighbour nodes are considered as valid context nodes. For example, for $v_i \in \mathcal{U}$, we have $N^t(v_i) \subseteq \{\mathcal{OD}, \mathcal{M}\}$. $p(n_j^t|v_i)$ is the conditional probability of observing type aware neighborhood $n_j^t \in G$ conditioned on the presence of v_i:

$$p(n_j^t | v_i) = \frac{e^{\mathbf{u}_j'^{\top} \cdot \mathbf{u}_i}}{\sum_{k=1}^{|V^t|} e^{\mathbf{u}_k'^{\top} \cdot \mathbf{u}_i}}, \qquad (2)$$

where $\mathbf{u}_j'$ is the context representation vector of v_j as a context node and $|V^t|$ is the number of nodes with type t in graph G. To reduce the computation complexity, we employ negative sampling [17] for efficient learning. The objective function becomes:

$$O_t = \log \sigma(\mathbf{u}_j'^{\top} \cdot \mathbf{u}_i) + \sum_{i=1}^{K} E_{v_n^t \sim U_n(v^t)} [\log \sigma(-\mathbf{u}_n^{t\top} \cdot \mathbf{u}_i)], \qquad (3)$$

where σ is the sigmoid function. The first term models observed edges in click events whereas the second term draws K negative edges from a uniform distribution. In this way, the distance between the learned user (resp. OD) embedding and each transportation mode embedding reflects the preference of a user (resp. OD) to each transport mode. That is, those users (resp. ODs) having similar transportation mode preference should be close to each other in the latent embedding space.

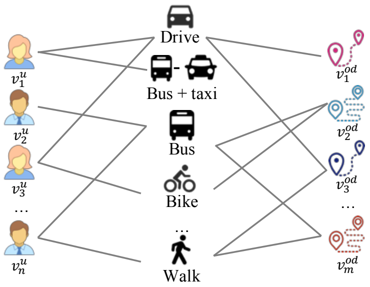


Figure 6: An illustrative example of the heterogeneous transportation graph. Each edge indicates the frequency of a user v_i^u (resp. OD pair v_j^{od}) clicking on a route of a specific transport mode.

3.5 Transportation Recommendation

We model the transport mode recommendation as a multi-class classification problem. Once the embedding vectors are learned, the proper transport mode can be derived by calculating the inner product of embedding vectors (as in [15]). In the production environment, however, the embedding method suffers from the cold-start problem. That is, 62.9% queries are from new users (*i.e.*, users migrate from other routing engines and new users of Baidu Maps) or target to new OD pairs (*i.e.*, OD pairs which have not been queried by users). To handle such cases, we concatenate the learned embedding vector of the user and the OD pair with the handcrafted features (as in Section 3.3) into a d dimensional feature vector.

Given a preprocessed dataset of n instances, m transport modes and d feature dimensions, we transform the raw data into a 2D matrix $\mathcal{D} = \{\mathbf{x}_i, y_i\}$ where $|\mathcal{D}| = n$, $\mathbf{x}_i \in \mathcal{R}^d$ is the feature vector and $y_i \in \mathcal{R}^{\mathcal{M}}$ is the i-th transport mode. We employ the gradient boosting tree [8] as our recommendation model because gradient boosting tree based algorithms [3] are suited for data mining with sparse and high dimensional features. Specifically, we sequentially generate a set of tree classifiers $\mathcal{F}(\cdot) = \{f_1(\cdot), f_2(\cdot), \ldots, f_k(\cdot)\}$ and ensemble the result of each classifier to generate the overall predictive result.

$$\hat{y}_i = \mathcal{F}(\mathbf{x}_i) = \sum_{j=1}^{k} f_j(\mathbf{x}_i), f_j \in \mathcal{F}, \tag{4}$$

where $\hat{y}_i$ is the estimated transport mode of i-th instance, $f(\cdot)$ is a softmax regressor for multi-class classification:

$$f(\mathbf{x}_i) = \frac{e^{\mathbf{w}_q^\intercal \mathbf{x}_i}}{\sum_{p=1}^{|\mathcal{M}|} e^{\mathbf{w}_p^\intercal \mathbf{x}_i}}, \tag{5}$$

where $\mathbf{w}_q$ is the parameter vector of the q-th class. The learning objective is to minimize

$$O = \sum_{i=i}^{n} l(y_i, \hat{y}_i) + \frac{\lambda_1}{2} \sum_{j}^{k} \|\mathbf{w_j}\|_1 + \frac{\lambda_2}{2} \sum_{j}^{k} \|\mathbf{w_j}\|_2, \tag{6}$$

where $l(\cdot)$ is is the cross-entropy loss, λ_1 and λ_2 are hyper-parameters for $L1$ and $L2$ regularizations, respectively.

The gradient of the tree function is derived much harder than traditional optimization tasks. Since we train classifiers sequentially, we approximate the gradient based on the previous step. The objective at the t-th iteration becomes

$$\widetilde{O}_i = \sum_{i=1}^{n} (g_i f_t(\mathbf{x}_i) + \frac{1}{2} h_i f_t^2(\mathbf{x}_i)) + \frac{\lambda_1}{2} \sum_{j}^{k} \|\mathbf{w_j}\|_1 + \frac{\lambda_2}{2} \sum_{j}^{k} \|\mathbf{w_j}\|_2, \tag{7}$$

where $g_i = \partial_{\hat{y}_i^{t-1}} l(y_i, \hat{y}_i^{t-1})$ and $h_i = \partial^2_{\hat{y}_i^{t-1}} l(y_i, \hat{y}_i^{t-1})$ are the first order and second order gradient statistics of $l(\cdot)$. The detailed deduction can be found in [7].

4 DEPLOYMENT

Hydra has been deployed on Baidu Maps. In this section, we describe the implementation and deployment details.

4.1 Offline Processing

Due to the complex data dependency, we propose an automatic pipeline for data integration and feature engineering. We employ Bigflow[1] as the offline data pipeline platform. Bigflow is an open source programming abstraction that allows for programming and processing data on various distributed computing engines (e.g., Hadoop Tez [19] and Spark [35]). In Bigflow, a set of data wrangling operators such as *map*, $filter$ and *join* is well supported and the lower level distributed operations are transparent to users.

4.1.1 Data pipeline. There are two components, *data preprocessing* and *feature construction*, in the data pipeline. In the data preprocessing phase, the user behavior data is appended on daily basis, the geographical data and user social data are updated on monthly basis and the meteorological data is collected per hour. In the midnight, each dataset is extracted from the log system of Baidu Maps and related databases. All datasets are integrated as described in Appendix D. The integrated dataset is stored as a large fact table. In the feature construction phase, features are extracted from the fact table. For numerical features, we replace missing values with either default values or the average value. Then we remove outliers and apply min-max normalization to scale the values into $[0, 1]$. For categorical features, we consolidate the rare categories into the "Other" category and then apply one-hot encoding to each categorical feature. Finally, all features and labels are combined into a two dimensional array and all side information, such as column min-max values and feature dimensions, is stored in a meta-data file.

4.1.2 Model Training. We use the XGBoost library[2] to train the recommendation model. The recommendation model is updated on daily basis to take new data into consideration. To exclude seasonal changes, we define a three-month sliding time window for training data selection. Once the data pipeline is finished, the model update script is triggered to update the model.

4.2 Online Processing

Baidu Maps answers billions of queries in each day. Thus, it is crucial to offer effective and scalable online service to users. To this end, we build efficient *region index* and scalable *web service framework* to enable low latency and high throughput online service.

[1] http://bigflow.baidu.com

[2] https://xgboost.readthedocs.io/en/latest/

4.2.1 Region Index. For online feature processing, a batch of statistical features is required to be mapped from coordinates to regions (e.g., join the origin coordinates of a query with the regional POI distribution). Traditional spatial index, such as R-tree, requires $O(\log n)$ search time, which is time consuming for cities with a large number of regions. We proposed a dedicated region index to speed up such mapping process. Specifically, we divide the city into fine-grained grids based on coordinates with a unique grid id. We then allocate regions to the corresponding grids. Note that each region is an irregular polygon, therefore, a grid may be intersected with one or multiple regions. For example, the minimum bounding rectangle (MBR) $[(116.30, 40.05), (116.31, 40.06)]$ is partitioned to grid g_1, with id 11630_4005. If there are two regions r_1 and r_2 intersect with g_1, the index in the database is stored as a key-value pair $(11630\_4005, [r_1, r_2])$, where the value is a list of regions. Internally, the grid-regions pair is stored as a hash table in Redis. Since the region is partitioned based on the road network, most grids are only associated with one or a few regions. In practice, the average time cost of the mapping process is much lower than that of R-tree.

4.2.2 Web Service Framework. We build the web service based on BRPC (https://github.com/brpc/brpc), a scalable RPC framework used throughout Baidu. The model is duplicated in four data centers distributed over China to reduce network latency of the service. Specifically, the online service contains three components. First, retrieve geographical information, meteorological data, user profile data in parallel and integrate them with raw route plans. Second, execute the online feature engineering process by leveraging the metadata generated in the offline data pipeline. Third, feed the processed feature vector into the model, sort each mode by model score and return the transport mode with the highest score to the user. About 6% of transport modes with the highest score have no corresponding plan. Instead, we recommend the next transport mode that has a feasible plan.

5 EXPERIMENTS

5.1 Experimental Setup

We conduct experiments on the datasets described in Section 2. We mainly focus on (1) the overall performance of our approach, (2) each feature contribution and (3) the robustness of our approach. We also present the user satisfaction analysis and the efficiency and scalability of our system. We split data from September 1 to November 20 as training set and the remaining as testing set.

Metrics. We adopt the overall NDCG [29], weighted precision, recall and F1 metrics to evaluate the performance. The NDCG metric takes all transport modes into consideration whereas the rest metrics only care about the top-1 recommendation.

Baselines. We compare our approach with two statistical recommendation methods and four learning based methods.

- **UHP** recommends the transportation mode of route using the fraction of user historical preference. The most common transport mode choice of the user will be recommended.
- **ODHP** recommends the transportation mode of route using the fraction of OD historical preference. The most popular transport mode between the OD pair will be recommended.
- **LR** recommends the transportation mode of route via the well-known logistic regression model. The input feature is same with our method as described in Section 3.3.
- **RF** recommends the transportation mode of route using Random Forest. The input feature is same to our method as described in Section 3.3.
- **LTR** is a popular LambdaMart [2] learning to rank method, where the pairwise loss is minimized. We use the plan feature described in Section 3.3 as input.
- **Trans2vec** is the state-of-the-art transportation mode recommendation method [15] based on graph embedding. It makes recommendation based on the inner product of user vector and transportation mode vector and the inner product of OD pair vector and transportation mode vector.

Table 2: Overall recommendation performance.

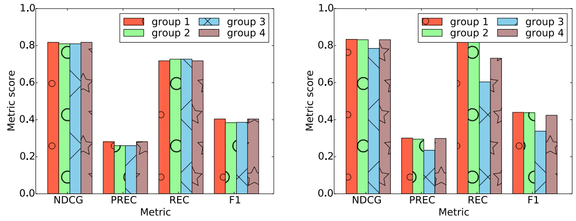
	Algorithm	Ndcg	Prec	Rec	F1
Beijing	UHP	0.29	0.159	0.207	0.18
	ODHP	0.343	**0.478**	0.229	0.31
	LR	0.802	0.255	0.681	0.371
	RF	0.754	0.329	0.448	0.379
	LTR	0.798	0.258	0.673	0.373
	Trans2vec	0.462	0.26	0.282	0.271
	Hydra	**0.815**	0.271	**0.72**	**0.396**
Shanghai	UHP	0.288	0.162	0.188	0.174
	ODHP	0.367	**0.454**	0.253	0.325
	LR	0.789	0.262	0.652	0.374
	RF	0.747	0.336	0.423	0.37
	LTR	0.794	0.265	0.653	0.377
	Trans2vec	0.46	0.266	0.258	0.262
	Hydra	**0.819**	0.274	**0.685**	**0.391**

Table 3: Top-10 features ranked by information gain.

Rank	Feature name	Relative gain
1	Walk ETA	1
2	Bus-cycle ETA	0.803
3	Bus ETA	0.577
4	Taxi-bus ETA	0.451
5	User walk percentage	0.295
6	Consumption level	0.213
7	Origin station count	0.162
8	Primary POI category	0.096
9	Hour	0.092
10	Spherical distance	0.051

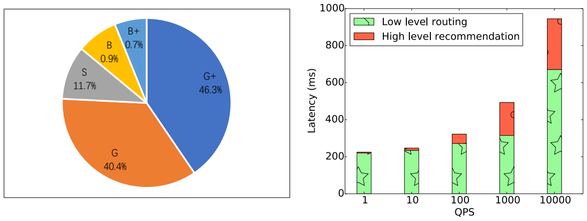
5.2 Overall Recommendation Result

Table 2 depicts the overall results of our method and all the compared baselines with respect to four evaluation metrics. We can make the following observations. *(i)* Hydra achieves better performance than six baselines over all metrics except Prec. Although the Prec of Hydra is worse than ODHP and RF, Hydra achieves better balance between Prec and Rec, which is evaluated by F1. *(ii)* The performance of LR is only slightly worse compared with Hydra and is competitive with LTR, which matches our expectation that situational context information and tailored feature engineering is curial for multi-modal transportation recommendation. *(iii)* The performance of solely Trans2vec is not well on the dataset with



(a) Group by users (b) Group by OD pairs

Figure 7: Robustness check on the Beijing dataset.



(a) User satisfaction (b) Latency

Figure 8: Results of the online service.

large proportion of cold-start users (resp. OD pairs). Overall, incorporating handcrafted features and high-order embedding features with a gradient boosting tree model outperforms all other baselines.

5.3 Feature Importance Analysis

To evaluate the effectiveness of feature construction, we rank features by information gain [16]. The higher information gain indicates higher frequency the feature used to split nodes in each individual tree. Table 3 reports top-10 features and their relative information gain. The top 4 features are all plan *ETA* of corresponding modes, which meets our expectation that travel time is the major consideration in the transport mode choice. Besides, we observe user attributes especially user social attributes such as historical mode preferences (walk preference in rank 5) and consumption level (rank 6) also make significant contribution for transport mode prediction. Features from rank 7 to rank 10 are spatial features and temporal features, which validates our intuition that the spatial and temporal dependency influences the transport mode choice.

5.4 Robustness Check

A robust algorithm should perform evenly on different subgroups of queries. We group queries from two perspectives: 1) user profile perspective, and 2) OD profile perspective. For 1), we segment users through gender and age, i.e., women and age lower than 35, men and age lower than 35, women and age older than 35, men and age older than 35. For 2), we segment OD pairs based on region functionality (i.e., we use the POI distribution of corresponding regions), classical K-means is applied to cluster OD pairs into four disjoint groups. Figure 7 illustrates the performance of our method on different subgroups on Beijing, the results on Shanghai are similar. For different groups of users, the results are strongly stable on four metrics, which validates the robustness of our method for different users. For different OD pairs, the results are also stable on four metrics expect the third group (e.g., for Rec, the difference is over 10%). This result indicates the variation from the OD profile perspective is more significant, further optimization on the third group can be applied in future to improve the overall performance.

5.5 User Interview

The model has been deployed on Baidu Maps since mid 2018. In past months, the model has answered over a hundred million route planning requests and served over ten million distinct users. To assess the user satisfaction of model recommendations, we published survey questionnaires to frequent Baidu Maps users. Overall, 738 valid questionnaires are collected. In the questionnaire, we set five level satisfaction categories, $G+, G, S, B, B+$, where G stands for good, S stands for same as before, B stands for bad. As shown in Figure 8(a), over 86.7% users think the recommendation result is better than before, and only 1.6% users think the recommendation result becomes worse. That is, our method provides better recommendations in terms of user experience.

5.6 Efficiency and Scalability

We further evaluate the query response latency of our framework in the production environment. The query response latency is composed of two parts, low level routing cost and high level recommendation cost. As reported in Figure 8(b), when we vary the query per second (QPS) from 1 to 10, 000, the low level routing latency increased from 220ms to 671ms, whereas the high level recommendation latency increased from 5ms to 274ms. Overall, the low level routing is the major bottleneck and can be further optimized. Note that the peak QPS of the online service is less than 1, 000, the online workload thus can be well handled by our system.

6 RELATED WORK

Route recommendation. Route recommendation has attracted much attention from both the academia (*e.g.*, [1, 28]) and industries (*e.g.*, Google Maps and Baidu Maps). A common routine of route recommendation is to apply the algorithms of shortest distance queries [9] with predefined cost functions [21]. As another important direction, the quality of recommended routes can be improved by leveraging large-scale historical trajectories [36]. Specifically, T-Drive [33] captures the intelligence of taxi drivers via a landmark graph. Dai *et al.* [4] recommends routes by considering personal preference (*e.g.*, time efficiency or fuel efficiency) for each individual driver. Zhou *et al.* [37] proposes a "semi-lazy" approach for path prediction. Recently, the route recommendation for shared mobility also attracted research interest to improve efficiency [27] and revenue [24–26]. However, all of them consider uni-modal route recommendations and thus cannot be directly applied for multi-modal route recommendation. Trans2vec [15] considers multi-modal recommendation by learning embedding of users, OD pairs and transport modes. But it suffers from the cold-start problem and requires extra models or strategies to handle new instances.

Urban computing. With the development of city urbanization, various data generated from GPS, sensors, buildings and humans has been applied to tackle various urban issues. For example, Yi *et al.* [31] and Yu *et al.* [32] predict urban safety by considering multiple spatial and temporal factors. Moreover, Tong *et al.* [23] and Xia *et al.* [30] predicts taxi demands based on multi-sourced urban data. Sun *et al.* [20] mines the urban region-of-interest through map search queries. As another example, Zhu *et al.* [38] captures user preferences for mobile recommendation. Motivated by above studies, we integrate multiple urban datasets to improve the performance of route recommendation among various transport modes. To the best of our knowledge, it is the first work that integrates multiple sources of urban data for route recommendation among various transport modes in a data-driven way at urban-scale.

7 CONCLUSION

In this paper, we presented Hydra, a personalized and context-aware multi-modal transportation recommendation system. It is a two-level system that adaptively recommends uni-modal and multi-modal transportation routes according to the user preferences and the situational context. We first extracted a rich set of features from user behavior data and several urban data collected from other sources. Next, we learnt embedding features via the heterogeneous transportation graph to enhance the recommendation performance. Moreover, a gradient boosting tree based model was devised for multi-modal transportation recommendation. Finally, we discussed several deployment issues to optimize Hydra to be scalable, including offline data pipelines, high performance spatial index, as well as the construction of web service framework. Extensive evaluations on real-world datasets validate the effectiveness and efficiency of Hydra. More importantly, Hydra has been deployed to Baidu Maps, one of the largest online map services, and has answered over a hundred million route recommendation queries made by over ten million distinct users.

ACKNOWLEDGMENTS

We would like to thank the Zhixing team of Baidu Maps for their valuable help and collaboration in the deployment process. Yongxin Tong's work is partially supported by National Science Foundation of China (NSFC) under Grant No. 61822201.

REFERENCES

[1] Hannah Bast, Daniel Delling, Andrew Goldberg, Matthias Müller-Hannemann, Thomas Pajor, Peter Sanders, Dorothea Wagner, and Renato F Werneck. 2016. Route planning in transportation networks. In *Algorithm engineering*. Springer, 19–80.

[2] Chris J.C. Burges. 2010. *From RankNet to LambdaRank to LambdaMART: An Overview*. Technical Report.

[3] Tianqi Chen and Carlos Guestrin. 2016. Xgboost: A scalable tree boosting system. In *SIGKDD*. ACM, 785–794.

[4] Jian Dai, Bin Yang, Chenjuan Guo, and Zhiming Ding. 2015. Personalized route recommendation using big trajectory data. In *ICDE*. 543–554.

[5] Daniel Delling, Julian Dibbelt, Thomas Pajor, Dorothea Wagner, and Renato F Werneck. 2012. *Computing and evaluating multimodal journeys*. KIT, Fakultät für Informatik.

[6] Julian Dibbelt et al. 2016. Engineering Algorithms for Route Planning in Multi-modal Transportation Networks. *Transportation* (2016).

[7] Jerome Friedman, Trevor Hastie, Robert Tibshirani, et al. 2000. Additive logistic regression: a statistical view of boosting. *The annals of statistics* 28, 2 (2000), 337–407.

[8] Jerome H Friedman. 2002. Stochastic gradient boosting. *Computational Statistics & Data Analysis* 38, 4 (2002), 367–378.

[9] Liping Fu, D Sun, and Laurence R Rilett. 2006. Heuristic shortest path algorithms for transportation applications: state of the art. *Computers & Operations Research* 33, 11 (2006), 3324–3343.

[10] Kenneth Gade. 2010. A non-singular horizontal position representation. *The journal of navigation* 63, 3 (2010), 395–417.

[11] Robert Geisberger, Peter Sanders, Dominik Schultes, and Daniel Delling. 2008. Contraction hierarchies: Faster and simpler hierarchical routing in road networks. In *International Workshop on Experimental and Efficient Algorithms*. 319–333.

[12] Andrew V Goldberg and Chris Harrelson. 2005. Computing the shortest path: A search meets graph theory. In *Proceedings of the sixteenth annual ACM-SIAM symposium on Discrete algorithms*. 156–165.

[13] Aditya Grover and Jure Leskovec. 2016. node2vec: Scalable Feature Learning for Networks. In *SIGKDD*.

[14] Yuxuan Liang, Songyu Ke, Junbo Zhang, Xiuwen Yi, and Yu Zheng. 2018. Geo-MAN: Multi-level Attention Networks for Geo-sensory Time Series Prediction.. In *IJCAI*. 3428–3434.

[15] Hao Liu, Ting Li, Renjun Hu, Yanjie Fu, Jingjing Gu, and Hui Xiong. 2019. Joint Representation Learning for Multi-Modal Transportation Recommendation. In *AAAI*.

[16] Gilles Louppe, Louis Wehenkel, Antonio Sutera, and Pierre Geurts. 2013. Understanding variable importances in forests of randomized trees. In *Advances in neural information processing systems*. 431–439.

[17] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*. 3111–3119.

[18] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. 2014. DeepWalk: online learning of social representations. In *SIGKDD*.

[19] Bikas Saha, Hitesh Shah, Siddharth Seth, Gopal Vijayaraghavan, Arun Murthy, and Carlo Curino. 2015. Apache tez: A unifying framework for modeling and building data processing applications. In *SIGMOD*. ACM, 1357–1369.

[20] Ying Sun, Hengshu Zhu, Fuzhen Zhuang, Jingjing Gu, and Qing He. 2018. Exploring the urban region-of-interest through the analysis of online map search queries. In *SIGKDD*. ACM, 2269–2278.

[21] Robert J Szczerba, Peggy Galkowski, Ira S Glicktein, and Noah Ternullo. 2000. Robust algorithm for real-time route planning. *IEEE Trans. Aerospace Electron. Systems* 36, 3 (2000), 869–878.

[22] Niraj Tolia, David G Andersen, and Mahadev Satyanarayanan. 2006. Quantifying interactive user experience on thin clients. *Computer* 3 (2006), 46–52.

[23] Yongxin Tong, Yuqiang Chen, Zimu Zhou, Lei Chen, Jie Wang, Qiang Yang, Jieping Ye, and Weifeng Lv. 2017. The Simpler The Better: A Unified Approach to Predicting Original Taxi Demands based on Large-Scale Online Platforms. In *SIGKDD*. 1653–1662.

[24] Yongxin Tong, Jieying She, Bolin Ding, Libin Wang, and Lei Chen. 2016. Online mobile Micro-Task Allocation in spatial crowdsourcing. In *ICDE*. 49–60.

[25] Yongxin Tong, Libin Wang, Zimu Zhou, Lei Chen, Bowen Du, and Jieping Ye. 2018. Dynamic Pricing in Spatial Crowdsourcing: A Matching-Based Approach. In *SIGMOD*. 773–788.

[26] Yongxin Tong, Libin Wang, Zimu Zhou, Bolin Ding, Lei Chen, Jieping Ye, and Ke Xu. 2017. Flexible Online Task Assignment in Real-Time Spatial Data. *PVLDB* 10, 11 (2017), 1334–1345.

[27] Yongxin Tong, Yuxiang Zeng, Zimu Zhou, Lei Chen, Jieping Ye, and Ke Xu. 2018. A Unified Approach to Route Planning for Shared Mobility. *PVLDB* 11, 11 (2018), 1633–1646.

[28] Sibo Wang, Wenqing Lin, Yi Yang, Xiaokui Xiao, and Shuigeng Zhou. 2015. Efficient route planning on public transportation networks: A labelling approach. In *SIGMOD*. 967–982.

[29] Yining Wang, Liwei Wang, Yuanzhi Li, and Di He. 2013. A theoretical analysis of NDCG ranking measures. In *COLT*.

[30] Yuan Xia, Jingbo Zhou, Jingjia Cao, Yanyan Li, Fei Gao, Kun Liu, Haishan Wu, and Hui Xiong. 2018. Intent-Aware Audience Targeting for Ride-Hailing Service. In *ECML*. Springer, 136–151.

[31] Fei Yi, Zhiwen Yu, Fuzhen Zhuang, Xiao Zhang, and Hui Xiong. 2018. An Integrated Model for Crime Prediction Using Temporal and Spatial Factors. In *ICDM*. 1386–1391.

[32] Zhiwen Yu, Fei Yi, Qin Lv, and Bin Guo. 2018. Identifying on-site users for social events: Mobility, content, and social relationship. *IEEE Transactions on Mobile Computing* 17, 9 (2018), 2055–2068.

[33] Jing Yuan, Yu Zheng, Chengyang Zhang, Wenlei Xie, Xing Xie, Guangzhong Sun, and Yan Huang. 2010. T-drive: driving directions based on taxi trajectories. In *SIGSPATIAL*. ACM, 99–108.

[34] Nicholas Jing Yuan, Yu Zheng, and Xing Xie. 2012. Segmentation of urban areas using road networks. *MSR-TR-2012–65, Tech. Rep.* (2012).

[35] Matei Zaharia, Mosharaf Chowdhury, Tathagata Das, Ankur Dave, Justin Ma, Murphy McCauley, Michael J Franklin, Scott Shenker, and Ion Stoica. 2012. Resilient distributed datasets: A fault-tolerant abstraction for in-memory cluster computing. In *NSDI*. USENIX Association, 2–2.

[36] Yu Zheng. 2015. Trajectory data mining: an overview. *TIST* 6, 3 (2015), 29.

[37] Jingbo Zhou, Anthony KH Tung, Wei Wu, and Wee Siong Ng. 2013. A "semi-lazy" approach to probabilistic path prediction in dynamic environments. In *SIGKDD*. ACM, 748–756.

[38] Hengshu Zhu, Enhong Chen, Hui Xiong, Kuifei Yu, Huanhuan Cao, and Jilei Tian. 2015. Mining mobile user preferences for personalized context-aware recommendation. *TIST* 5, 4 (2015), 58.

A EXTENDED DATA DESCRIPTION

We first introduce three kinds of user behavior data in detail, include *query record*, *display record* and *click record*.

- **Query record.** A query record consists of a session ID, an anonymized user ID, a time stamp, the coordinates and the POIs of the origin o and the destination d, and the operating system of the device. For example, $[s_i, u_i$, "$2018-09-01\ 15:15:36$", $(116.30, 40.05)$, $(116.353, 39.99)$, "*Baidu Building*", "*Beihang University*", "*IOS*"] means a user u_i makes a query on a trip from Baidu Building to Beihang University in the afternoon of September 1st, 2018.
- **Display record.** A display record consists of a session ID, an anonymized user ID, a time stamp and a list of routes. Each route consists of the transport mode, the estimated route distance, the estimated time of arrival (ETA), the estimated price and the rank in the display list. The number of displayed routes varies across queries, and there can be no feasible routes for certain queries.
- **Click record.** A click record consists of a session ID, an anonymized user ID, a time stamp, and a list of clicked routes in the route list. There can be none or multiple clicks on a route. We only record the first click on each route and remove repeated clicks.

Then we present three kinds of geographical data in detail, include *POI data*, *road network data* and *transportation station data*.

- **POI data.** Semantics in POIs indicate the travel intention and have been applied for various urban computing tasks [14]. Our POI dataset contains $1,204,344$ distinct POIs in Beijing and $1,594,684$ distinct POIs in Shanghai. Each POI record has a POI ID, an ascendant POI ID, coordinates of the location, the POI name and a two-level category. The ascendant POI is the higher level POI of the current POI. For example, "Baidu building" is the ascendant of "Baidu building tower 2". To enrich the POI semantics, we map uninformative POI categories such as "Entrance" and "Door Address" to the ascendant POI categories. The two-level category has a primary category and a secondary category. For example, "Education" is a primary category whereas "University" is one of its secondary categories. There are 18 primary categories and 189 secondary categories in the POI dataset.
- **Road network data.** Road network data help to capture regional traffic capability. Each record of road network consists of a unique road segment ID, the start location coordinates, the end location coordinates, the road length and the level of the road segment. There are eight levels of road segments. For instance, the national highway is with the highest level and the pedestrian path is with the lowest level.
- **Transportation station data.** The distribution of transportation stations also influences user preferences on transportation modes. For regions with few bus stations, taxis might be preferred. Each record of transportation station data consists of a unique station ID, coordinates of the station location, a list of bus lines across the station and the corresponding city code.

Table 4 shows the distribution of primary POI categories. The spatial distribution of POIs in Beijing is show in Figure 9(a). Figure 9(b) shows the spatial distribution of road networks and transportation stations, where the yellow lines are road segments and black points are stations. Similar to user activities (as shown in Figure 2(a) and Figure 2(b)), the density of POIs, road segments and bus stations in the urban central area is much higher. As described in Section 2.3, Figure 9(c) shows the distribution of weather in each day. Overall, there are more rainy days in September and November whereas more sunny days in October.

Table 4: Statistics of Primary POI Categories of Beijing.

ID	Category	Count
P01	Residence	163,733
P02	Shopping	137,882
P03	Company	137,223
P04	Entrance	110,667
P05	Life Service	85,448
P06	Food	78,088
P07	Government	37,546
P08	Education	35,035
P09	Beauty	19,650
P10	Healthcare	16,123
P11	Finance	14,688
P12	Entertainment	13,894
P13	Hotel	12,700
P14	Culture Venue	9429
P15	Sports	9,022
P16	Tourist Attraction	7,763
P17	Door Address	7,709
P18	Administrative area	4,069

B DETAILED FEATURE LIST

Table 5 is the feature list used in Hydra, including *Plan features*, *Spatial features*, *Temporal features*, *Meteorological features*, and *User features*.

C SPHERICAL DISTANCE CALCULATION

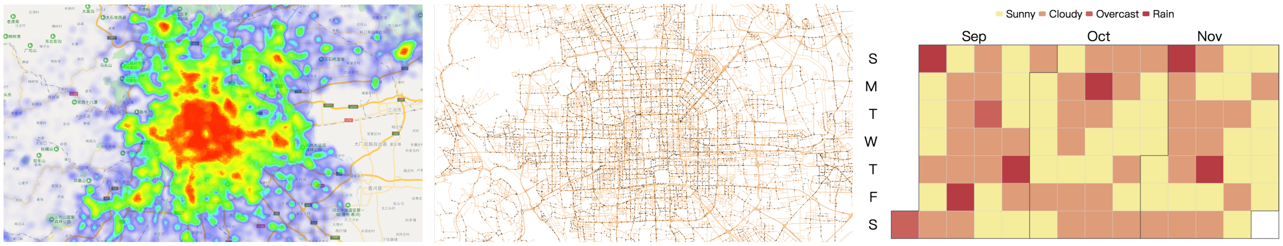
Given (φ_1, λ_1) as the coordinates of origin o and (φ_2, λ_2) as the coordinates of destination d. The spherical distance of od is calculated as follows:

$$\begin{aligned} d_{od} =& 2R \cdot atan2\Big(\sqrt{\sin^2(\Delta\varphi_{od}/2) + \cos\varphi_1 \cos\varphi_2 \sin^2(\Delta\lambda_{od}/2)}, \\ & \sqrt{1 - \sin^2(\Delta\varphi_{od}/2) - \cos\varphi_1 \cos\varphi_2 \sin^2(\Delta\varphi_{od}/2)}\Big) \end{aligned} \tag{8}$$

Where $\Delta\varphi_{od} = \varphi_1 - \varphi_2$ and $\Delta\lambda_{od} = \lambda_1 - \lambda_2$. We set $R = 6371$ to approximate the distance of od on the earth's surface.

D DATA INTEGRATION

We integrate multi-source urban datasets into a unified dataset to create a more comprehensive view of transportation mode choices. Specifically, by integrating the user behavior data and the meteorological data, we can find the key weather factor influencing users' transport mode preference. As another example, by integrating the



(a) POI distribution.

(b) Road network and station distribution.

(c) Weather distribution.

Figure 9: More distributions of the Beijing dataset: (a) the distribution of POIs; (b) the distribution of road networks and stations; (c) the distribution of weather

Table 5: The Description of features.

Feature Type	Feature	Description
Plan	Road network distance	The length of the planned route on the road network
	Price	The total cost of the plan
	ETA	The estimated time of arrival (ETA) of the plan
	Transfer count	The number of transfers of the plan
	Transport mode count	The number of transport modes used in the plan
Spatial	District	The administrative district which the origin and destination belongs to
	POI category	The primary and secondary category of the POI
	Spherical distance	The spherical distance between the OD pair
	Station distance	Spherical distances of top-k nearest bus stations from the O/D location
	Station count	The number of bus stations in the O/D region
	Regional POI distribution	The distribution of two level POI category of corresponding O/D region
	Regional road network distribution	The number of road segment and road intersection in the O/D region
	Regional bus line count	The number of bus line cross the O/D region
	Regional historical mode distribution	The mode preference distribution of O/D/OD region
Temporal	Hour	The corresponding time period in a day
	Minute	The corresponding minute bin
	Day of week	The ordinal number of the day in a week
	Day of month	The ordinal number of the day in a month
	Workday	Whether the day is a workday
Meteorological	Weather	The weather in current time period
	Temperature	The temperature and statistics (i.e., highest/lowest temperature) in current day
	AQI	The AQI and AQI statistics (i.e., highest/lowest AQI) in current day
	Wind speed	The wind speed in current time period
	Wind direction	The wind direction in current time period
User	Demographic attribute	The age, gender of the user and OS in use
	Social attribute	The education level, industry type, car type and consumption level
	User historical mode	The mode preference distribution of the user

user behavior data and POI data, we can analyze the relationship between transport modes and travel intention.

We use the *JOIN* operator to integrate all datasets together. We first join user queries, display and click records on the session ID. Since each query contains an origin and a destination, we further join the origin and the destination with the POI data through location coordinates and POI names. Note that in map search queries, 91% origins are "current locations", which do not have explicit POI names. For such origins, we associate the coordinates with the nearest POI. Besides, the meteorological data is in district level. Therefore, we join the origin and the destination with the meteorological record if the coordinates located in the corresponding district. We crawl the polygon of each district as a preprocessing step and then join the original and destination with the meteorological record if the coordinates located in the corresponding district. Finally, we join the above data with user profile data through user IDs.