Context-aware Multi-data Fusion to Achieve Finest Ride-share Schedule Using Big Data

Avinash Potluri¹ and N.V. Narendra Kumar²
IDRBT¹,², Hyderabad
E-mail: potluri.avinash1@gmail.com¹

Abstract. Enabling the taxi rides to share empty seats in their vehicles, to riders, is a win-win situation for both riders and drivers. Existing methods mainly target on minimizing the cost of travel for vehicles. In this work we focus on assigning a new rider to a vehicle considering the real-time context of both riders and drivers. The crux of the problem lies in analysing massive amounts of data in an instance of time to obtain the optimal ride schedule. We further study the scenario, to accommodate best compatible riders with minimal change in the route, to maximize the satisfaction of all possible entities in the business model in several aspects. Our system takes advantage from route recommendation and utility maximization approaches. We also inherit the benefits of collaborative filtering to provide best ride experience. The effectiveness of our approach is shown with experimentation on various datasets. Our work has significant importance, as it redefines the personalization preferences based on real-time context.

1. Introduction

From the traditional way of hailing a taxi service there has been a significant change because of the ridesharing benefits provided by several taxi-service brands in the industry. Ridesharing facility has gained huge interest in short time from all communities because of its one stop solution, providing eco-friendly and economical ride, reducing the traffic congestion. There can be a large scale adoption to this service once the autonomous vehicles come into place. Indeed, this has become the vision of several tech companies like Tesla, Waymo and Uber. With the estimate of having massive scale business these companies started to fine tune the utilization of available resources.

With the vast development of innovative technologies, individuals’ every day life has a notable change. From online shopping to ride-sharing applications, there is a vast difference, compared to the traditional way of performing the same activity. In this paper, we address the practical problem in ridesharing, namely context-aware ridesharing, which assigns riders to the available vehicles considering the real-time context of both rider and driver. The objective is to maximize profit, minimize the detours and maximize the satisfaction of all possible entities in the business model. For instance, consider the following example, where we expect to have riders of similar context in a ride.

Assume the capacity of the vehicle is three and there are two such vehicles and five distinct people waiting for a ride with approximately the same source and destination. Out of the five people three are going for an interview, whereas the other two are going for a movie. Our work focuses on designing...
two ride schedules, where, in one ride schedule all the people going for interview are accommodated and in the other ride schedule the two people going for movie are accommodated.

The example shown above is completely different from the existing studies. We aim at improving the satisfaction of the users with best context relevant ride schedule. Given a set of users u1, u2, u3, u4, u5, ...u_n and vehicles v1, v2, ...v_m the objective of our system is to recommend the association among users and vehicles, considering and assuring the context. The system has to return a best relevant setting in the available subspace. We drive around the issue of suggesting a viable intuitive solution for the ride-sharing e-business application. In this work, we utilize the available information to compute the end goal of improving the satisfaction of ride-sharing travelers. As figured, the key issue should be scheduling a route. With every request having a source and destination, existing works on ride-sharing has concentrated mainly on two viewpoints: accommodating all requests and minimizing the travel cost for the ride.

The following are the specific contributions in this paper:
- Develop an algorithm that recommends best path from all possible paths that can accommodate maximum number of passengers into the vehicle.
- We formulate a problem that recommends appropriate passenger into the vehicle based on real-time context.
- We overcome the computational bottleneck in computing while considering the contexts of both riders and drivers that redefines the personalization preferences.
- In addition we our method, maximizes the satisfaction of riders and drivers with minimal change in waiting time, route and cost.

The following is the organisation of the paper. In section 2 the overview of the problem and the need to explore this is emphasised. Section 3 deals with existing and proposed approaches. Section 4 shows the experimental results followed by conclusion and future work in section 5.

2. Overview of Problem Statement
The formulation of our problem comes from the following scenarios, Route Recommendation, Valid Scheduling for Dynamic Resources, Driver Interests, Similarity Matching and Pooling based on Context. We initially project the scope and need for formulating this problem. The overview of the problem with an example for each scenario is given below:

2.1 Route Recommendation
The ride sharing scenario where the shortest route is selected based on the initial rider, there is a chance of having a financial loss if the taxi does not find a customer in that route. The challenge is to recommend a route such that there is a maximum chance of accommodating the riders with minimum cost. This becomes more complex in the case where the riders should also have similar interests. The best suitable route is selected from the all possible routes available. Though we are greedy of getting the best route we need solve the issue in an instance of time. Consider figure 1, the nodes numbered 1 to 7 are nodes, where we have a ride request and a to h is cost incurred in that route. Assume the initial request is at 1 and the destination is at 7. Since, the shortest path is via 1, 4 and 7, the vehicle can accommodate only one more rider. But, if the route recommendation engine suggests another route i.e. either 1, 2, 3 and 7 or 1, 5, 6 and 7 we can accommodate three riders into the vehicle. Analysing all these possible routes for a ride is having significant role in maximizing the profit.
2.2 Valid Scheduling for Dynamic Resources

To accommodate the riders to a vehicle we need to compute. Our structure to store data for computing a ride schedule is as follows. We have start, maximum time to detour, schedule update and end. Firstly, the initial ride request, its location, start time and destination and end time. The maximum time to detour is given, where, we can update the current schedule without deviating much, the existing riders schedule. This is given as a schedule update to the driver. This depends on the number of available requests at that instance of time and the number of available vehicles.

![Schedule](image)

2.3 Driver Interests

The selection of a vehicle and indeed the driver should match the interest of riders. For instance, if the language is common among the driver and the riders, the ease of clarifying address issues can be easily resolved. Similarly if the last ride taken by the driver is near his home, the satisfaction of the driver also increases. Our focus also lies in improving the satisfaction of riders as well as drivers. As shown in figure 3 D1 and D2 are the current locations of drivers with their vehicles. Let A, Band C be the riders requests. D1H and D2H are the homes of drivers. For D1, the schedule can be D1, A, C, D1H or D1, B, C, D1H and for D2, the schedule can be D2, A, C, D2H or D2, B, C, D2H. In contrast to the existing systems, which focus only on riders interests, our work also focus on drivers interest, and allocates the schedule which has minimum distance from their home from their last destination.

![Driver Scenario](image)

2.4 Similarity Matching

The naive strategy to select riders in a vehicle is to select in the order in which the request is made in that route without exceeding the capacity of vehicle. But, to achieve maximum satisfaction from the riders we need to solve a bigger puzzle. With the constraint of effectively meeting the supply and demand we need to select the best matching riders in a vehicle. The matching can be starting with gender to the educational level. To evaluate the best match we can find the riders who have common friends. Similarity matching can be calculated based on the figure 4. Let A and B be two different riders, their common interests can be compared with a score for each feature in C1, C2 and soon CN.
From the available requests, we compute the score and the best matched score is used as one of the weights in deciding the riders in a vehicle.

### 2.5 Pooling based on Context

To select the best compatible riders in a pool ride, the preferences of the users need to be known prior. Establishing correlation among the riders can guide us in selecting the appropriate riders. The preferences of users can be captured in several ways. Social media is one such place where the user interests can be highly captured. The huge availability of user interests like favorite color, interest in the kind of movies, favorite sports, favorite hero, favorite tourist place etc. in social media applications allows us to discriminate the riders. Though, the preferences obtained from social media benefits us, the optimal preference lies in the real-time context of the rider. We include the context-aware feature, which adds firmness in grouping the riders in a pool.

For instance, what can be the optimal selection..? whether selecting a group of users who are completely new to place or associating both new and old. By analysing the queries based on context-aware situations we can find common patterns. For example, the purpose of taking a ride can be going to a political meeting, hospital, mall, movie, competition, interviews, local trip, hotel, enjoy good weather, taking children to school and going to college etc. We can also benefit, if we observe the change of interests over time and contexts. Consider figure 5, there are five ride requests with approximately nearer source and destination. The context of visit for rider 1, 3 and 5 is interview and its same for all of them. The context of visit for rider 2 is movie and rider 4 is mall. As rider 2 and 4 is similar, the riders 2 and 4 has to be accommodated into the same vehicle. Riders 1, 3 and 5 need to be accommodated into same vehicle, which may improve the overall satisfaction of the system.

![Fig. 5 Context Scenario](image)

### 3 Approaches

As discussed above, we need to respond to the requests of the riders. We consider a set of ride requests and set of vehicles available in an instance of time. Before the context based rider allocation representation is given, we discuss some naive approaches from the literature. We cover naive rider assignment approach, greedy approach and context-aware approach.

#### 3.1 Naive Rider Assignment Approach

There are several approaches for allocating rider to a vehicle. In general, the rider allocation system, for a given request raised from a user the system designs a schedule. This is based on the source,
destination and the instance of time. Usually this will be the shortest path. The path that needs to be traversed to pick-up and drop-off is assigned and shared to the driver and the rider. Similarly, for all riders. If the system gets other rider requests in the initial scheduled route, the system reschedules the route without deviating much from the initial schedule and accommodates the other riders. The algorithm shown in 1 describes the naive rider assignment approach. We assume there are a set of ride requests and set of vehicles available in that instance of time.

Algorithm 1 Naive Rider Assignment
1: Input : rider r i and Vehicle v j
2: Output : A schedule for rider and vehicle
3: T s : = { T s | s = 1, 2, 3, ..., s } where each T s is a valid schedule
4: T t : = { T t | t = 1, 2, 3, ..., t } where each T t is a valid schedule at an instance of time t from riders source and destination.
5: Selection of rider and update the schedule by traversing all possible schedules.

3.2 Greedy Approach
In the process of improving the naive approach of assigning a rider to a vehicle, a greedy approach has been developed. In this approach we tend to have a parameter tuning option, where, we can tune the system based on our interests. If our interest is to maximize the profit, we can tune the system accordingly. If our interest is to minimize the time, the system can be tuned accordingly. We can even select the best compatible customers into the vehicle. If we are greedy to match the interests of the users, we obtain the social connections from social networking sites and find the common interests of the riders. We find the best score among the riders combination and design a schedule. This can also be done based on the common language they speak. For instance we design a schedule with A, B and C with driver d1 and D, E and F with driver d2 considering the common language of drivers and riders. The system benefits the parameter to which we are greedy. In practice we can change to suitable parameter to which we are greedy, as the set of ride requests and set of vehicles available in that instance of time change dynamically.

Algorithm 2 Greedy Approach
1: Input : available rider requests R and available vehicles V
2: Output : A schedule for rider and vehicle
3: T t : = { T t | t = 1, 2, 3, ..., t } where each T t is a valid schedule at an instance of time t from riders source and destination.
4: Retrieve all possible vehicles { V i | x = 1, 2, 3, ..., v }
5: For each vehicle V i , find the best combination of riders { R j | j = 1, 2, 3, ..., r } such that the cost is minimized.
6: For the selected rider vehicle combination, update the schedule S

3.3 Context-aware Solution - Our Approach
The proposed approach is a context based solution for assigning rider to a vehicle. This approach overcomes the drawbacks of traditional methods, which focus on maximization of profit, minimization of detour etc. In contrast to these techniques our method groups the riders in a single schedule considering the real-time context. The input to our method is set of riders who raised request of a ride and set of vehicles available in that instance of time. v z is the vehicle available, r x is the rider available, ru u is a possible route from the given source to destination and c s is the context of rider / driver. Initially a request is considered and based on the source and destination, a nearest vehicle is assigned. This is an initial schedule. Now, based on the available pending requests which are near to the initial source and destination, all possible routes are analyzed and a new route is rescheduled without having much deviation from the initial schedule. Let the difference be α time, β is the distance. Based on the context of rider in the vehicle, each of the other requests are given a score.
From the all possible routes, a route which has the chance of accommodating maximum number of riders is selected. Similarly, the similarity scores are also calculated based on the common interests of the riders and drivers. Each of these parameters is given a weightage. Let the weightage be $w_a$, $w_b$, $w_c$. These weights can be tuned accordingly based on the requirement of the system. Once the weights are finalized, the scores are sorted and the schedules are designed. These schedules consists of the riders, who have similar context as well as interests. This guarantees the increase in the overall satisfaction of the system. Finally, a fine tuned route schedule is given to the riders, calculated from all possible combinations, such that the satisfaction score is maximized.

**Algorithm 3 Rider-Route Assignment**

1: Input : rider $r_x$ and vehicle $v_z$
2: Output : Finest schedule for rider and vehicle
3: $V_z := \{ V_z \mid z = 1, 2, 3, ..., z \}$ where $v_z$ is the vehicle available at an instance of time $t$.
4: $R_x := \{ R_x \mid x = 1, 2, 3, ..., x \}$ where $r_x$ is the rider available at an instance of time $t$.
5: $RU_u := \{ RU_u \mid x = 1, 2, 3, ..., u \}$ where $ru_u$ is a possible route from source to destination.
6: $C_s := \{ C_s \mid s = 1, 2, 3, ..., d \}$ where $c_s$ is the context of rider / driver $r_x$.
7: Selection of Route - RU such that the maximum capacity of vehicle is equalized.
8: Compute the score of satisfaction among the riders $R_x 1, R_x 2, ..., R_x x$ in vehicle $V_z$.

$$\text{combined score: } R(x_1) ./ R(x_2) ./ R(x_3) ./ ... ./ R(x_x)$$

9: Fine tuning the combination of routes and riders such that the satisfaction score is maximized.

In our approach, we use the intelligence of Collaborative Filtering to find the best match. From the origin in 18th century, till today there has been extensive research in improving the accuracy and performance of Collaborative Filtering techniques. The taxonomy of autoencoders using Maximum-likelihood estimation in a regular autoencoder evolves from:

$$L_{\beta}(x_u; \theta, \phi) \equiv E_{q_\phi(z_u|x_u)}[\log p_\theta(x_u|z_u)] - \beta KL(q_\phi(z_u|x_u)||p(z_u))$$

$$\theta^{AE}, \phi^{AE} = \arg\max_{\theta, \phi} \sum_{x_u} \log p_\theta(x_u|\phi(x_u))$$

In this scenario, our focus is not on optimizing the Collaborative Filtering technique, but rather we used one of the best available Collaborative Filtering techniques [Liang et al.(2018)Liang, Krishnan, Hoffman, and Jебara]. The route recommendation is inferred from [Huang et al.(2014)Huang, Bastani, Jin, and Wang]. The combination of Variational Autoencoders for Collaborative Filtering and route recommendation with context information improves the overall satisfaction of the system.

**4 Experiments**

We used synthetic data to show our context-aware approach. We consider a set of ride requests and set of vehicles available in an instance of time. The user interests are categorized into different types and are considered for finding similarities.

The graphs shown below are the satisfaction scores obtained for the synthetic datasets in various settings. The improvement in the satisfaction of the users highlight the importance of context.
Fig. 6 Satisfaction Score

Fig. 7 Satisfaction Score

Fig. 8 Satisfaction Score

5 Conclusion
In this paper, we develop a Context-aware Multi-data Fusion approach using collaborative filtering and route recommendation system. Our approach enables to go beyond the traditional models capacity which minimizes cost. We empirically show through experimentation that, considering real-time context improves the satisfaction of users in the system. This paper best suits the current ride-sharing needs and in future when there is a large scale adoption to the autonomous vehicles. Our context-aware approach recommends the best compatible riders in a vehicle, capturing the real-time context, similarity and common interests with minimal change in time and route. In future one can explore the best way to capture the information since the real-time context and route recommendation are completely data-driven.