International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 06 | June - 2023 SJIF Rating: 8.176 ISSN: 2582-3930

Location Based Recommendation System for Car Reservation Using Machine Learning

Pooja Deshpande[1], Dipti Shiwankar[2], Neha Patil[3], Anjali Pansare[4], Sagar Shinde[5]

[1234]Student, Department of Computer Engineering, Modern Education Society's College of Engineering Pune – 411001, Maharashtra, India.
[5]Assistant Professor, Department of Computer Engineering, Modern Education Society's College of Engineering Pune – 411001, Maharashtra, India.

ABSTRACT

The increasing popularity of car-sharing services has led to a growing need for effective recommendation systems that can suggest the most suitable car rental options based on user preferences and location. This paper presents a location-based recommendation system for car reservation that utilizes machine learning techniques to enhance the user experience and optimize the car rental process. The proposed system leverages a combination of user data, vehicle availability, and geographic information to generate personalized recommendations. Machine learning algorithms, such as collaborative filtering and content-based filtering, are employed to analyze historical user behavior and preferences, as well as vehicle features and characteristics, to make accurate and relevant recommendations. The system considers various factors, including the user's location, travel itinerary, preferred car types, and historical rental patterns, to provide tailored recommendations. By utilizing real-time data, such as traffic conditions and car availability, the system can also adapt its recommendations dynamically to ensure up-to-date and reliable suggestions.

Keywords: Recommender system, Location-based social network, Place recommendation, Cab booking, User based collaborative filtering, Machine learning, Prediction, Algorithms.

I. INTRODUCTION

With the rise of on-demand car rental services and the increasing availability of shared mobility options, users often face the challenge of finding the most convenient and suitable vehicle for their needs. Traditional methods of searching for car rentals can be time-consuming and inefficient, leading to suboptimal choices and user dissatisfaction. This is where a location-based place recommendation system comes into play.

In recent years, the use of machine learning (ML) techniques has significantly transformed various industries and revolutionized the way we interact with technology. One such application is the development of location-based place recommendation systems. These systems leverage ML algorithms to provide personalized recommendations for car reservations based on a user's specific location.

By harnessing the power of ML, these systems can analyse a vast amount of data, including historical rental information, user preferences, and real-time location data, to offer accurate and customized recommendations. The primary goal is to match users with the most suitable

car rental options based on their current location, travel requirements, and individual preferences.

The ML algorithms employed in these recommendation systems employ techniques such as collaborative filtering, content-based filtering, and hybrid approaches to generate accurate and personalized suggestions. Collaborative filtering techniques leverage the wisdom of the crowd, making recommendations based on the preferences of similar users. Content-based filtering, on the other hand, focuses on analysing the characteristics and features of the available cars and matching them with the user's preferences.

The key advantages of using a location-based place recommendation system for car reservations are efficiency, convenience, and enhanced user experience. By taking into account the user's current location, the system can provide real-time recommendations, eliminating the need for extensive searches and reducing decision-making time. Additionally, by considering user preferences and past rental history, the system can deliver highly personalized suggestions, ensuring a higher level of customer satisfaction.

The introduction highlights the concept of a location-based place recommendation system for car reservations using machine learning (ML). It explains that traditional methods of searching for car rentals can be inefficient and time-consuming. To address this, ML algorithms are employed to analyse data such as historical rental information, user preferences, and real-time location data. The introduction mentions that ML algorithms utilize collaborative filtering, content-based filtering, and hybrid approaches to generate accurate recommendations. Collaborative filtering considers the preferences of similar users, while content-based filtering focuses on the characteristics and features of available cars. The combination of these techniques ensures personalized suggestions for users.

A. Problem Statement

Based on previous surveys, users / customers found it difficult to travel to new places without knowing locality, people and places. It gets difficult for them to search and explore places on their own. Searching for places to visit and searching for guides is also a point of concern. At the same time, tourist guides also find it difficult to gather customers. Existing cab booking systems don't have place recommendation and tourist guide booking all in one place. So, we need a system which has both of these features which will be convenient for users to find services integrated in a single system. By addressing these issues, the proposed system aims to enhance the user experience, optimize car rental decisions, and promote efficient utilization of car-sharing services.

B. Motivation

1) If a tourist visits any particular place for the first time, then it is difficult to explore nearby places.
2) At the same time finding a tourist guide is troublesome.
3) So, building a system which include all these features will overcome such problems.

C. Objectives

1) Identify the symbolic expression through images/videos.
2) Communication gap can be easily bridged.
3) Eliminate the need of a translator.

II. LITERATURE SURVEY

Based on previous surveys, users / customers found it difficult to travel to new places without knowing locality, people and places. It gets difficult for them to search and explore places on their own. Searching for places to visit and searching for guides is also a point of concern. At the same time, tourist guides also find it difficult to gather customers. Existing cab booking systems don't have place recommendation and tourist guide booking all in one place. So, we need a system which has both of these features which will be convenient for users to find services integrated in a single system.

The very first method of the project includes cloning of the uber app into our local machine. For this purpose, we used ReactJs, NextJs, Firebase, Python, TensorFlow, dataset, Map Box and various other tech stacks. The basic functionality of how uber works in real time was implemented on our local platform.

The user requests a cab service from the app, sets the destination and the cab arrives at the user's pick up point.

Map box uses a variety of algorithms and techniques to find the shortest path between two destinations on its mapping platform. The specific algorithm used depends on the context and requirements of the routing problem. One commonly used algorithm is the A* (A-star) algorithm, which is a popular pathfinding algorithm in graph theory. Map box's routing engine takes into account various factors when calculating the shortest path, such as road network data, traffic conditions, turn restrictions, and other considerations. By considering these factors, the algorithm can generate accurate and optimized routes for various modes of transportation, including driving, walking, and cycling.

Dijkstra's algorithm is a popular algorithm used to find the shortest path between nodes in a graph. It guarantees the shortest path if all edge weights are non-negative. Here's how Dijkstra's algorithm works: 1) Initialize the algorithm: Create a set of unvisited nodes and mark all nodes as unvisited. Assign a tentative distance value of infinity to all nodes, except the starting node, which is assigned a distance value of 0.Set the starting node as the current node. 2) Visit the neighbors: For the current node, consider all of its unvisited neighbors and calculate their tentative distances from the starting node. Compare the newly calculated tentative distance to the current assigned value and update it if it's smaller. 3) Mark the current node as visited and select the unvisited node with the smallest tentative distance as the next current node. Go back to step 2. 4) Repeat step 2 and 3 until all nodes have been visited or if the smallest tentative distance among the unvisited nodes is infinity. In this case, there is no connection between the initial node and the remaining unvisited nodes.5) Once all nodes have been visited or there are no more unvisited nodes, the algorithm terminates. The shortest path from the starting node to each node is determined by following the path with the smallest tentative distance.

The A* algorithm is a widely used heuristic search algorithm that finds the shortest path between two nodes in a graph. It combines the best features of Dijkstra's algorithm (uniform-cost search) and a heuristic function to guide the search towards the goal node more efficiently.

The efficiency and effectiveness of the A* algorithm heavily rely on the accuracy of the heuristic function. The heuristic function should provide an estimate of the remaining cost from each node to the goal node without overestimating it (i.e., it should be admissible). A common heuristic used is the Euclidean distance or Manhattan distance between the node and the goal node in a grid-based graph.

The A* algorithm is widely used in various applications, including pathfinding in games, route planning in maps, and optimization problems. It efficiently explores the graph by prioritizing nodes with lower total cost, guided by the heuristic information, to find the shortest path from the starting node to the goal node.

Predicting cab prices using machine learning involves creating a model that can learn patterns and relationships from historical data to make predictions on future cab prices. Here's a general outline of the steps involved in building a machine learning model for cab price prediction: 1) Data Collection: Gather historical data related to cab rides, including features such as distance traveled, duration of the trip, pickup and drop-off locations, time of day, day of the week. 2) Data Preprocessing: Prepare the collected data for training the machine learning model. 3) Feature Engineering: Extract or create additional features from the available data that can provide more insights and improve the model's predictive power. 4) Model Selection: Choose an appropriate machine learning algorithm for cab price prediction. Some commonly used algorithms for regression tasks include linear regression, decision trees, random forests, gradient boosting 5) Model Training: Train the selected machine learning model using the prepared training data. 6) Model Evaluation: Evaluate the trained model's performance using the testing data.
It's important to note that the success and accuracy of the cab price prediction model depend on the quality and representativeness of the data, as well as the careful selection of features and algorithm.

Regression: Regression models are commonly used for price prediction tasks. Linear regression is a simple yet effective technique for predicting the future values of a

target variable based on historical data. Other types of regression models, such as polynomial regression, decision tree regression, and random forest regression, can also be used for price prediction. Here are the steps that can be followed:

1) Data Collection
2) Data Cleaning and Pre-processing
3) Feature Selection
4) Model Selection
5) Train the Model
6) Model Tuning
7) Deployment

By following these steps, we can build a machine learning model that can accurately predict the prices of cab bookings based on various factors, which can improve the user experience of the cab booking application.

Our project will be integrated using various recommendation system for a user-friendly experience. The recommendations for places are done based on the popularity-based recommendation. This provides suggestions based on the ratings of the places. The highly-rated place is considered to be the most popular place by the engine and provides them as the recommendation to the users. This system doesn't require any prior information like the browsing of places, suggestions preferred user history, or the details of the places. The only factor that is considered under this is the ratings provided to provide a scalable recommendation system.

Another recommendation system is content-based filtering, which generates predictions by analyzing item attributes and searching for similarities between them. Here, the system uses features and likes in order to recommend with things that user might like. It uses the information provided by user over the internet and the ones they are able to gather and then they curate recommendations according to that.

The system can also provide tourists with a local guide who can show them around the city and provide information about the local culture, history, and attractions. The tourist guide can be an add-on service that the user can opt for while booking the cab.

The system can also collect data about user behavior, cab utilization, and tourist preferences. This data can be analyzed to improve the system's efficiency, optimize cab routing, and provide personalized recommendations to users.

III. METHODOLOGY

Existing cab booking systems don’t have place recommendation and tourist guide booking all in one place. So we need a system which has both of these features which will be convenient for users to find services integrated in a single system.

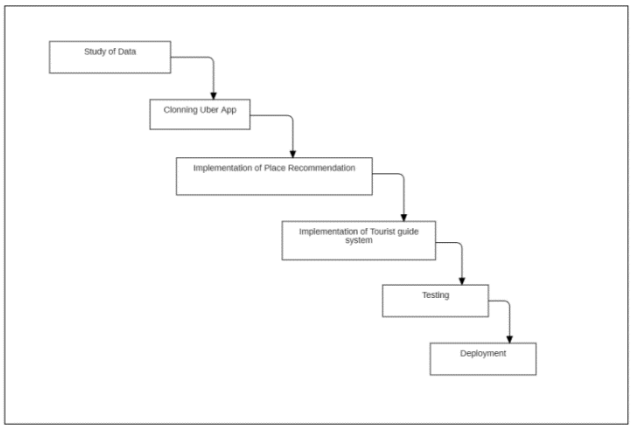

Fig.1 Flow Diagram

customer can login to system with user ID and password and if new to system, then can register to system with email. After successful login customer can explore places, they know and can get recommendation from the system. If the customer is new to place and didn't know much about place then this system will help them to find the new places with highest rating, good reviews etc.
Along with place recommendation this system will also provide tourist guide (driver)to customer if they need. After confirming place by customer, if driver is available for ride, then he will accept the request and generated OTP will be send to customer’s side. According to distance covered by customer payment amount will get generate and customer can pay by various payment gateways available.

After the trip, the system would prompt the user to provide feedback and ratings for the cab driver and tourist guide. This feedback can be used to improve the service and provide better recommendations to future users.

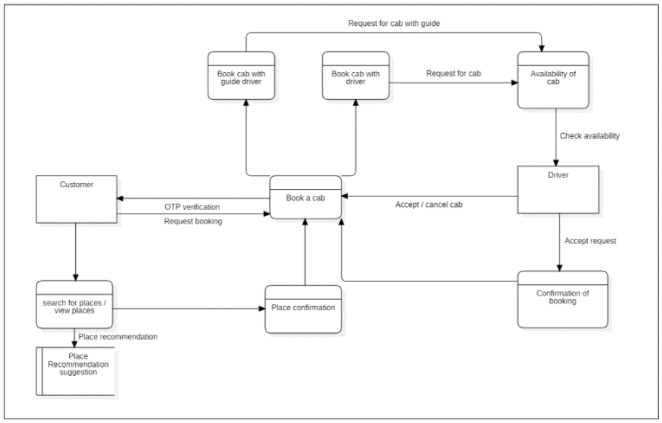

Fig.2 Block Diagram

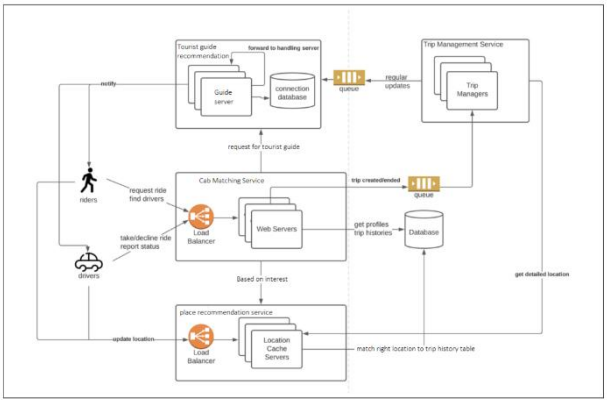

Fig.3 Architecture Diagram

System contain various services like cab matching, trip management, payment gateways, place recommender as well as tourist guide recommender with database storage. This service matches ride requests with available drivers based on proximity, driver rating, and other factors. Payment service handles payment processing, including credit card payments, and handles refunds and chargebacks. Trip Management service handles the lifecycle of a trip, including pickup, navigation, and drop-off. This service collects and analyzes data from various sources to provide insights into rider behavior, driver behavior, and operational efficiency.

Database stores all the data related to rides, drivers, passengers, payments, and other aspects of the platform in a highly scalable and fault-tolerant data store.

System is Build using many technologies. we use ReactJs and NextJs which are JavaScript framework to build our frontend it is recently high in demand and easy to implement. Map box is the main API which we use to handle operation associate with map. we use python to build our backend part. Recommender system is build in Deep Neural Network which is part of machine learning. After building all this modules we integrate it with the help of flask and Django so we can get expected recommendations.

IV. RESULT AND DISCUSSION

This is the User-Interface of our Project for creating or logging to the application. So, User can login to the system by using their mail-id. They can create account and do login to this application.

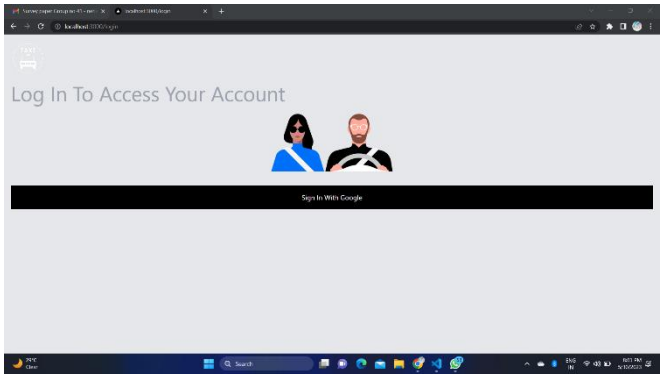

Fig. 4 Login Page

After filling the required information and clicking to the Sign in with Google button, user profile will be generated and can successfully logged-in to the system

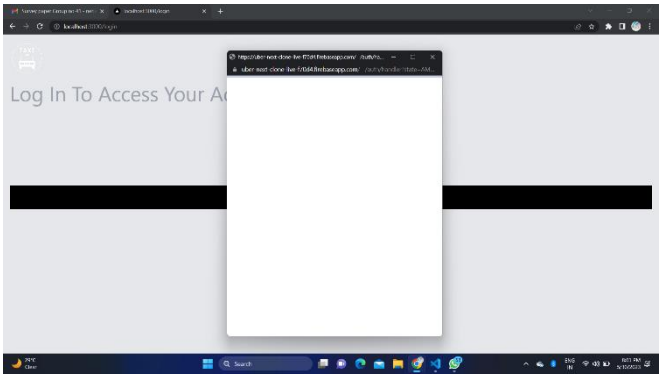

Fig 5. Account Logging in

After successful logging to the application, user will get this front page for booking and exploring more car options like car, auto, etc.

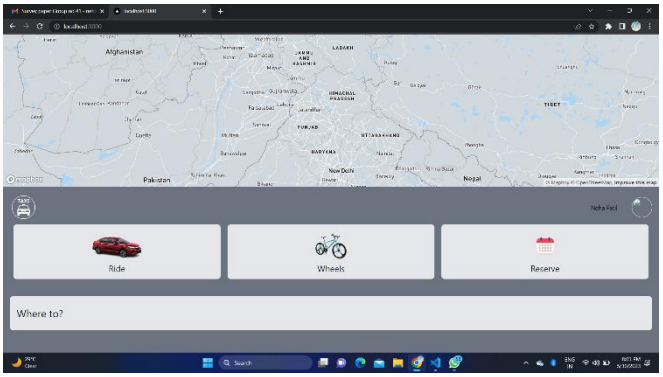

Figure 11.3: Home Page

After clicking on where to button the user can add/search respective pick-up and drop-up destinations in the searching bar. After adding the locations the user can confirm the places and also explore the recommended places

.

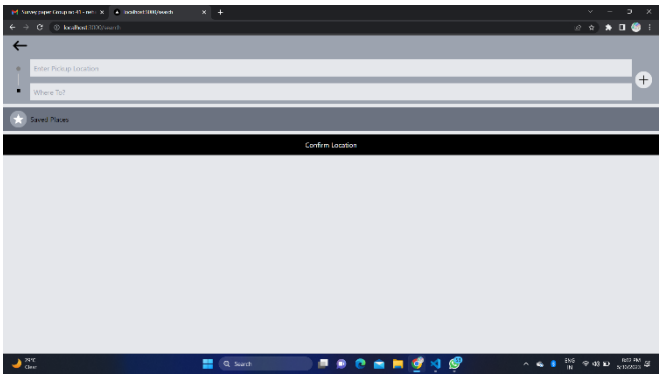

Figure 11.4: Search Bar

The nearby recommended/suggested places will be generated based on the ratings based on the user's choices.

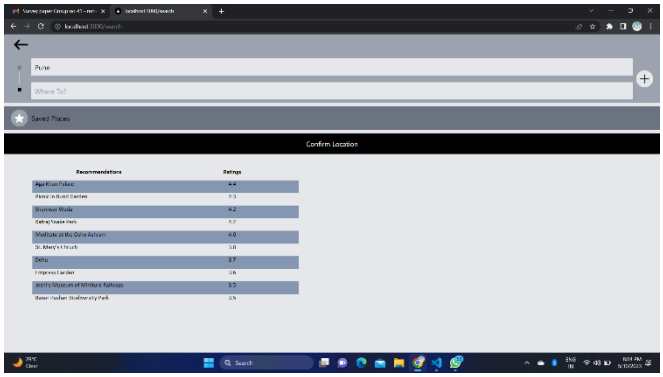

Figure 11.6: Recommendation near place

Places between the selected route will get recommend to user according to the ratings

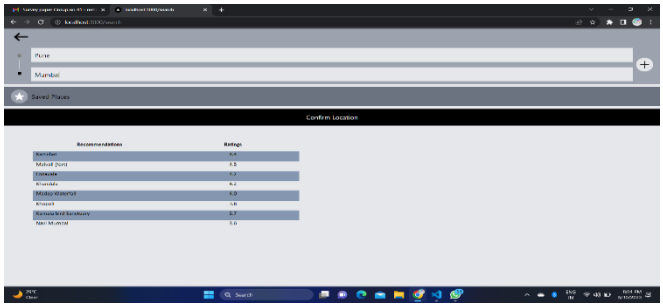

Figure 11.7 Recommendation in Route

The pick-up and drop-off points will be pointed out on the map and the fare, time, distance will be generated. Also, the places which are in between the route will also displayed on the map. So, the user can explore or visit them accordingly

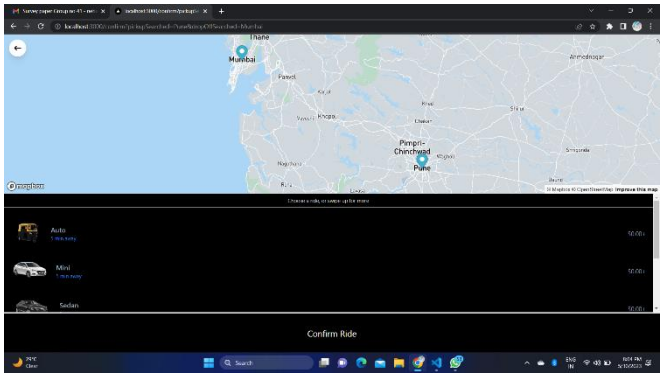

Figure 11.5: Ride Selection

V. CONCLUSION

Customers can use an online cab booking system to book/rent cabs efficiently. Customer can find his/her profile, browse cabs and book them using this system. The bulk of people rely on cab services for their daily transportation needs. Cab booking system offered by many transportation companies in many cities but it lacks somewhere. it is a system in which integration of two system is going to take place i.e., cab booking with place recommendation. Place recommendation will help people to explore new and highly rated places without taking advice from any third-party application. It will be more beneficial for local and tourist as well. This paper demonstrates an effective cab booking system.

VI. REFERENCES

[1] Rishi Srinivas, B. Ankayarkanni 2, R.Sathya Bama Krishna 3 "Uber Related Data Analysis using Machine Learning", Proceedings of the Fifth International Conference on Intelligent Computing and Control Systems (ICICCS 2021) IEEE Xplore Part Number: CFP21K74-ART; ISBN: 978-0-7381-1327-2

[2] Elizabeth Rani. G; Sakthimohan. M; Revanth Raj. R; Sri Ganesh. M; Shyam Sunder. R; Karthigadevi. K "An Automated Cost Prediction in Uber/Call Taxi Using Machine Learning Algorithm" 2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE) IEEE

[3] Ummesalma M; Yashiga C "COLPOUSIT: A Hybrid Model for Tourist Place Recommendation based on Machine Learning Algorithms" 2021 5th International Conference on Trends in Electronics and Informatics (ICOEI) IEEE

[4] Kavita Agrawal, K. Maldanna, G. Nalin Raj "Taxi Demand Prediction System Using Machine Learning" Volume IX, Issue VI, JUNE/2020

[5] Elizabeth Paskahlia Gunawan; Cuk Tho "Development of an Application for Tourism Route Recommendations with the Dijkstra Algorithm" 2021 International Conference on Information Management and Technology (ICIMTech) IEEE

[6] Ms. Priya Naik,Ms. Palak V. Desai ,Ms. Supriya Patil "Location Based Place Recommendation using Social Network" 2019 5th International Conference for Convergence in Technology (I2CT) Pune, India. Mar 29-31, 2019

[7] KEYAN GAO, XU YANG, CHENXIA WU, TINGTING QIAO, XIAOYA CHEN, MIN YANG, AND LING CHEN "Exploiting Location-Based Context for POI Recommendation When Traveling to a New Region Exploiting Location-Based Context for POI Recommendation When Traveling to a New Region" IEEE

[8] Q. Xu, J. Wang and B. Xiao, "Personalized Location Recommendation for Location-Based Social Networks," IEEE/CIC International Conference on Communications in China (ICCC), October 2017.

[9] Swaroopa V Dugani, Sunanda Dixit, Mahesh Belu "Automated Adaptive Sequential Recommendation of Travel Route" Proceedings of the IEEE 2017 International Conference on Computing Methodologies and Communication (ICCMC)

[10] Eriko Shibamoto; Chalisa Kittirojrattana; Chawan Koopipat; Aran Hansuebsai; Kosuke Takano "A Recommendation System of Sightseeing Places based on User's Behavior of Taking and Editing Photos "2019 IEEE Pacific Rim Conference on Communications, Computers and Signal Processing (PACRIM) IEEE