Mobility service analysis using machine learning in Python

[1.]Harshal S. Hemane, [2.] Ansh Sahu, [3.] Ankush Singh, [4.] Anshuman Raj

[1]Assistant Professor, Bharati Vidyapeeth (Deemed to be University) College of Engineering, Pune, India
[2,3,4]Student, Bharati Vidyapeeth (Deemed to be University) College of Engineering, Pune, India

---***---

Abstract -

In the contemporary era of urbanization, mobility services have become integral to daily life, offering convenient transportation solutions. This study presents a comprehensive analysis of mobility services, leveraging the power of machine learning techniques. Specifically, sentiment analysis, time series analysis, and multinomial Naive Bayes (NB) methodology are employed to delve into the intricacies of user experiences and service performance.

The sentiment analysis component explores the sentiment polarity of user reviews and feedback pertaining to mobility services. By employing natural language processing (NLP) techniques, sentiment trends are identified, offering insights into customer satisfaction, concerns, and preferences. These sentiments serve as valuable indicators for service providers to enhance user experiences and address areas of improvement.

Furthermore, time series analysis is employed to discern patterns and trends in mobility service usage over time. By analyzing historical data, seasonality, trends, and anomalies are identified, enabling forecast models to predict future service demand and optimize resource allocation. This temporal perspective provides valuable foresight for service providers to efficiently manage fleets and infrastructure.

Additionally, the multinomial Naive Bayes methodology is applied to classify user sentiments and feedback into distinct categories. By leveraging probabilistic algorithms, this approach categorizes sentiments into predefined classes, such as positive, negative, or neutral. This classification enables a deeper understanding of user perceptions and facilitates targeted interventions to improve service quality.

Through the integration of sentiment analysis, time series analysis, and multinomial Naive Bayes methodology, this study offers a holistic framework for analyzing and optimizing mobility services. By leveraging machine learning techniques, service providers can gain actionable insights to enhance user satisfaction, operational efficiency, and overall service performance. This research contributes to the advancement of data-driven decision-making in the realm of urban mobility, fostering sustainable and inclusive transportation ecosystems.

Key Words: sentiment analysis, Multinomial Naive Bayes (NB), Time Series analysis, Natural Language Processing

1.INTRODUCTION

In an era marked by rapid urbanization and technological advancements, mobility services have emerged as fundamental components of modern urban landscapes. These services, encompassing ridesharing, bike-sharing, public transit, and more, play a pivotal role in facilitating convenient and efficient transportation solutions for individuals and communities alike. However, amidst the proliferation of mobility options, ensuring optimal service quality, user satisfaction, and operational efficiency presents ongoing challenges for service providers.

To address these challenges, this study embarks on a comprehensive analysis of mobility services, harnessing the capabilities of machine learning techniques. In particular, the integration of sentiment analysis, time series analysis, and multinomial Naive Bayes (NB) methodology offers a multifaceted approach to unraveling the complexities inherent in user experiences and service dynamics.

The escalating volume of user-generated content, including reviews, feedback, and social media interactions, provides a rich source of data for understanding user sentiments towards mobility services. Leveraging sentiment analysis techniques, this study seeks to extract valuable insights from unstructured textual data, shedding light on the varying degrees of user satisfaction, concerns, and preferences.

Moreover, the temporal dimension of mobility service usage is explored through time series analysis, which aims to uncover underlying patterns, trends, and fluctuations in service demand over time. By analyzing historical data, this approach enables the development of predictive models to anticipate future demand patterns, facilitating proactive resource allocation and service optimization.

Furthermore, the application of multinomial Naive Bayes methodology offers a probabilistic framework for classifying user sentiments and feedback into distinct categories. Through this classification process, sentiments are categorized as positive, negative, or neutral, providing a structured framework for understanding user perceptions and informing targeted interventions for service improvement.

By synthesizing these methodologies, this study endeavors to provide a comprehensive framework for analyzing and optimizing mobility services. Through the lens of machine learning, service providers can gain actionable insights to enhance user satisfaction, operational efficiency, and overall service performance. Ultimately, this research contributes to the advancement of data-driven decision-making in urban mobility, paving the way for sustainable and inclusive transportation ecosystems in an increasingly connected world.

2. LITERATURE SURVEY

Sentiments, the fundamental expressions of human emotions, serve as crucial cues akin to feature extraction in problem-solving scenarios. This study focuses on sentiment analysis using Twitter data concerning cab services provided by Uber and Ola. By scrutinizing user tweets, it aims to discern their needs and complaints, offering insights for service enhancement.

Deep learning techniques, including Google Word2Vec, are employed to extract information from tweets effectively. By categorizing tweets into positive and negative sentiments, the study utilizes two deep learning algorithms, Deep Feed Forward Neural Network and Convolutional Neural Network, trained on labeled data to gauge accuracy. Testing data is then used to assess the algorithms' predictive capabilities. Python is the chosen programming language for its versatility and extensive libraries. Through this approach, the study aims to leverage deep learning and sentiment analysis to improve cab services and user experiences by addressing user concerns expressed on Twitter.

This study aims to delve into predictive analysis, a key methodology within Machine Learning. Numerous companies, such as Ola and Uber, leverage Artificial Intelligence and machine learning techniques to tackle challenges like accurate fare prediction. After conducting a comparative analysis of regression and classification algorithms, this paper proposes a methodology to enhance prediction modeling for more precise fare estimations. The research caters to stakeholders involved in fare forecasting, addressing the evolving complexities of cab fare determination. While traditional fare estimation relied solely on distance, technological advancements have expanded the factors influencing fares, including time, location, passenger count, traffic, duration, and base fare. Grounded in supervised learning, particularly prediction applications in machine learning, this study contributes to refining fare prediction models for improved accuracy and efficiency in the transportation sector. abbreviations in the title or heads unless they are unavoidable.

After the text edit has been completed, the paper is ready for the template. Duplicate the template file by using the Save As command, and use the naming convention prescribed by your conference for the name of your paper. In this newly created file, highlight all of the contents and import your prepared text file. You are now ready to style your paper.

3. SOFTWARE

The list of components used: -

3.1 Visual Studio code: Visual Studio Code (VS Code) stands as a premier code editor renowned for its versatility, efficiency, and extensive feature set. Developed by Microsoft, this lightweight yet powerful tool has gained immense popularity among developers across various programming languages and platforms. Its intuitive user interface, coupled with robust language support, enables seamless coding experiences for professionals and enthusiasts alike. With features like IntelliSense, Git integration, and debugging capabilities, VS Code streamlines the development workflow, boosting productivity and collaboration. Moreover, its vast ecosystem of extensions allows users to customize their environments to suit specific needs, ranging from language support and code formatting to project management and cloud integration. Whether developing web applications, mobile apps, or cloud services, VS Code empowers developers to write, debug, and deploy code with unparalleled ease and efficiency, making it an indispensable tool in the modern software development landscape

.**3.2 Python Language:** -Python is a versatile and powerful programming language known for its simplicity and readability, making it an ideal choice for beginners and experienced developers alike Python, revered for its simplicity and versatility, has become the cornerstone of modern programming. Its clean syntax and extensive libraries make it ideal for a wide range of applications, from web development to scientific computing. Python's dynamic nature allows for rapid prototyping and iteration, facilitating innovation and problem-solving across industries. Its vibrant community fosters collaboration and continual improvement, ensuring that Python remains at the forefront of technology. Whether you're a seasoned developer or a novice enthusiast, Python's accessibility and power make it the language of choice for tackling complex challenges with elegance and efficiency.

4. LIBRARY USED

4.1 NumPy:- NumPy, short for Numerical Python, stands as the cornerstone of scientific computing in Python. Its powerful array-processing capabilities facilitate efficient manipulation of large datasets and complex mathematical operations. NumPy provides a high-performance multidimensional array object along with tools for working with these arrays. Its extensive library of mathematical functions enables seamless integration with other scientific computing libraries like SciPy and Pandas, forming a robust ecosystem for data analysis and manipulation. NumPy's efficient implementation in C and Fortran ensures speedy execution, making it indispensable for tasks ranging from basic array operations to advanced linear algebra and statistical computations. As an essential component of the Python data science stack, NumPy continues to empower researchers, engineers, and data scientists in their quest for insights and innovation.

4.2 Pandas:- Pandas, a Python library built on top of NumPy, revolutionizes data analysis and manipulation with its intuitive data structures and powerful tools. At its core lies the DataFrame, a two-dimensional labeled data structure akin to a spreadsheet or SQL table, offering flexibility and ease of use for handling structured data. Pandas excels in data wrangling tasks, allowing users to clean, transform, and merge datasets effortlessly. Its rich functionality includes powerful indexing, reshaping, and grouping operations, enabling complex data manipulations with just a few lines of code. Additionally, Pandas seamlessly integrates with other Python libraries, such as Matplotlib and Scikit-learn, facilitating streamlined workflows for data visualization and machine learning. Whether exploring data, conducting statistical analysis, or preparing data for modeling, Pandas empowers users with the tools to extract meaningful insights and drive informed decision-making in diverse domains, from finance and healthcare to academia and beyond.

4.3 Matplotlib :- Matplotlib, renowned as the foundational plotting library in Python, empowers users to create high-quality visualizations with ease. Its versatility allows for the creation of a wide array of plots, ranging from simple line charts to complex heatmaps and 3D plots. Matplotlib's intuitive interface and extensive customization options provide users with fine-grained control over every aspect of their plots, ensuring that visualizations are tailored to specific needs and preferences. Whether for exploratory data analysis, presentation-ready graphics, or publication-quality figures, Matplotlib's robust functionality and compatibility with various data formats make it an indispensable tool for scientists, engineers, analysts, and educators alike. Moreover, its active community and comprehensive documentation ensure ongoing support and continual innovation, solidifying Matplotlib's position as the go-to choice for data visualization in the Python ecosystem.

4.4 Seaborn :- Seaborn, built on top of Matplotlib, serves as a high-level interface for creating attractive and informative statistical graphics in Python. With its concise syntax and powerful abstraction layer, Seaborn simplifies the process of generating complex visualizations from structured data. It provides a wide range of plot types, including scatter plots, histograms, box plots, and heatmaps, each designed to reveal patterns and relationships within the data. Seaborn's emphasis on aesthetics ensures that plots are not only visually appealing but also convey insights effectively. Furthermore, Seaborn integrates seamlessly with Pandas DataFrames, allowing for seamless data manipulation and visualization. Whether exploring relationships between variables, visualizing distributions, or highlighting trends over time, Seaborn empowers users to create insightful and publication-quality graphics with minimal effort, making it a valuable asset for data scientists, analysts, and researchers across disciplines.

4.5 OS Library :- The `os` library in Python serves as a powerful utility for interacting with the operating system, offering a wide range of functions for managing files, directories, and processes. With `os`, developers can perform tasks such as navigating file systems, manipulating paths, and executing system commands, regardless of the underlying platform. This library provides an abstraction layer that simplifies platform-specific operations, ensuring code portability across different operating systems. Developers can use `os` to create, delete, or modify files and directories, check file permissions, and retrieve information about the system environment. Additionally, `os` facilitates process management, allowing for the creation and manipulation of system processes, as well as handling environment variables and signals. Whether automating file operations, managing system resources, or executing system-level tasks, the `os` library offers a comprehensive suite of functions to streamline interaction with the underlying operating system, enhancing the versatility and robustness of Python applications.

4.6 SciKit :- The scikit-learn library, often referred to as scikit, stands as a cornerstone of machine learning in Python, offering a comprehensive suite of tools for building and deploying predictive models. With its user-friendly interface and extensive documentation, scikit-learn democratizes machine learning by making complex algorithms accessible to developers and researchers alike. From classification and regression to clustering and dimensionality reduction, scikit-learn provides implementations of a wide range of machine learning algorithms, along with utilities for data preprocessing, model evaluation, and hyperparameter tuning. Moreover, scikit-learn's emphasis on code readability and consistency fosters collaboration and accelerates the development of machine learning solutions. Its seamless integration with other Python libraries, such as NumPy, Pandas, and Matplotlib, further enhances its usability and versatility. Whether for academic research, industrial applications, or personal projects, scikit-learn empowers users to harness the power of machine learning for solving real-world problems with confidence and efficiency.

5. METHODOLOGY

5.1 Sentiment Analysis :- Sentiment analysis is a valuable tool in mobility services analysis projects employing machine learning techniques. By leveraging sentiment analysis algorithms on textual data such as user reviews, social media posts, and customer feedback, mobility service providers can gain insights into user satisfaction, preferences, and sentiments towards their services. This analysis can help identify areas for improvement, understand user perceptions of service quality, and tailor offerings to better meet customer needs. For instance, sentiment analysis can categorize user reviews as positive, negative, or neutral, enabling providers to prioritize addressing issues highlighted by negative feedback and reinforce positive aspects of their services. Furthermore, sentiment analysis can aid in monitoring brand reputation and identifying emerging trends or concerns among users, allowing providers to proactively respond to customer sentiment and maintain a positive brand image. By integrating sentiment analysis with machine learning models, mobility service providers can enhance customer satisfaction, increase loyalty, and drive continuous improvements in their services.

5.2 Time Series Analysis ;- Time series analysis plays a crucial role in analyzing mobility services data using machine learning algorithms. In such projects, time series data often includes information like ride volumes, durations, geographic locations, and user behavior patterns collected over time. By applying time series analysis techniques, such as trend analysis, seasonality decomposition, and forecasting, to this data, machine learning models can uncover valuable insights and make informed predictions about future mobility trends. For instance, time series analysis can help identify patterns in ride demand throughout the day, week, or year, allowing mobility service providers to optimize fleet management and resource allocation. Additionally, it can assist in detecting anomalies or unexpected changes in ride patterns, which may indicate issues like traffic congestion, service disruptions, or changes in user preferences. By integrating time series analysis with machine learning algorithms, mobility service providers can enhance operational efficiency, improve service reliability, and ultimately deliver better experiences for users.

5.3 Multinomial NB :- In projects analyzing mobility services using machine learning, the Multinomial Naive Bayes (MultinomialNB) algorithm can be a valuable tool for various tasks. MultinomialNB is particularly useful for text classification tasks, making it suitable for analyzing textual data such as user reviews, feedback, or customer support tickets in the context of mobility services. By representing each text document as a bag-of-words or TF-IDF vector, MultinomialNB can learn the probability distribution of words across different categories or sentiments (e.g., positive, negative, neutral). This enables the algorithm to classify new texts into the appropriate category based on their word frequencies, allowing mobility service providers to automatically categorize user feedback or sentiment analysis results. For example, MultinomialNB can classify user reviews as positive, negative, or neutral, providing insights into customer sentiment towards specific aspects of the mobility service. By integrating MultinomialNB into the analysis pipeline, mobility service providers can automate the categorization of textual data, enabling faster decision-making, targeted improvements, and enhanced customer satisfaction.

5.4 Text Analysis :- Text analysis plays a pivotal role in projects analyzing mobility services through machine learning techniques. By extracting insights from textual data sources such as user reviews, social media comments, or customer support tickets, text analysis enables mobility service providers to understand customer sentiments, preferences, and concerns. Natural Language Processing (NLP) techniques are employed to preprocess and analyze the text, including tasks such as tokenization, part-of-speech tagging, and sentiment analysis. For instance, sentiment analysis algorithms can categorize user feedback as positive, negative, or neutral, providing valuable insights into user satisfaction and areas for improvement. Additionally, topic modeling techniques like Latent Dirichlet Allocation (LDA) can identify recurring themes or topics within the textual data, helping providers understand common issues or interests among users. Furthermore, text analysis can facilitate personalized recommendations and targeted marketing campaigns by analyzing user preferences and behavior patterns expressed in text data. By leveraging machine learning models for text analysis, mobility service providers can enhance customer experience, improve service quality, and drive business decisions based on actionable insights extracted from textual data.

6. RESULTS and DISCUSSION

6.1 Results

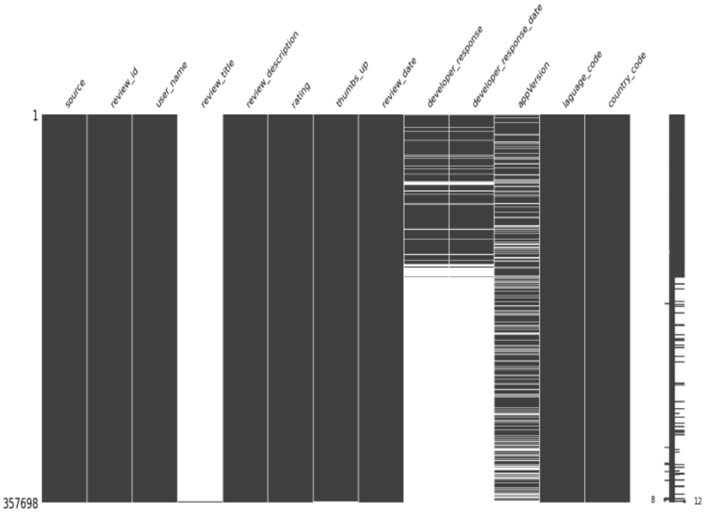

Figure-1: Matrix OLA

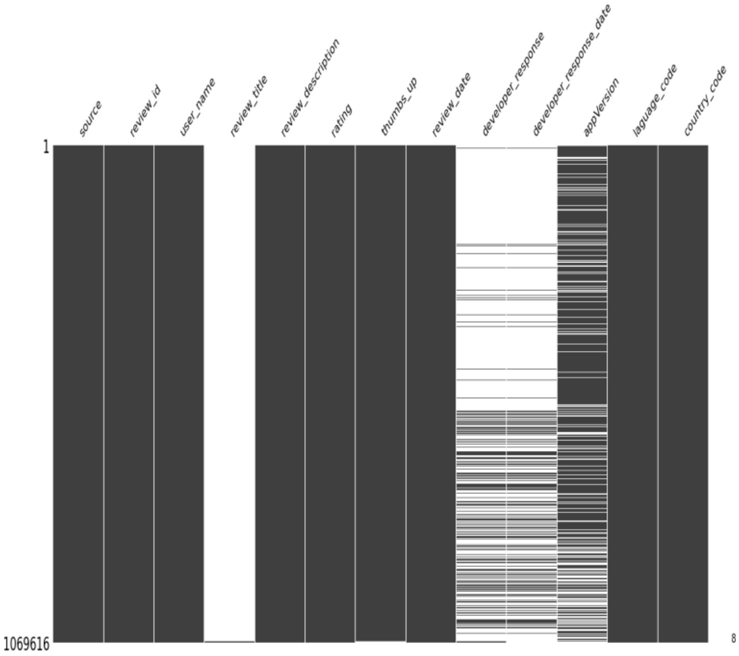

Figure-2: Matrix UBER

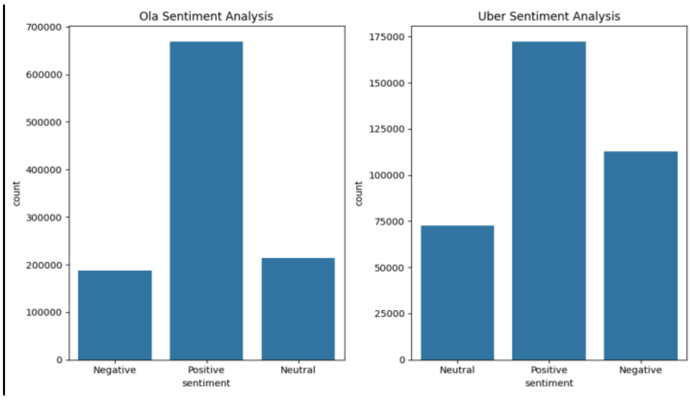

Figure-3: Text Analysis

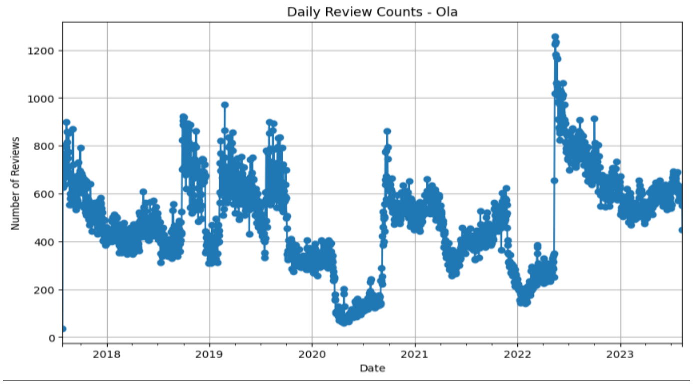

Figure- 4: time series analysis of ola

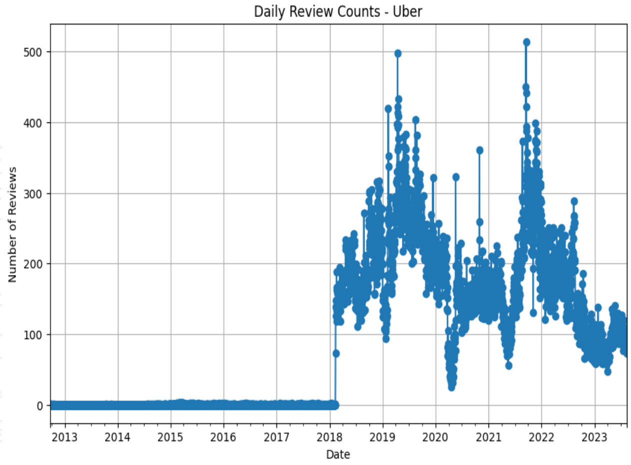

Figure-5: time series analysis of Uber

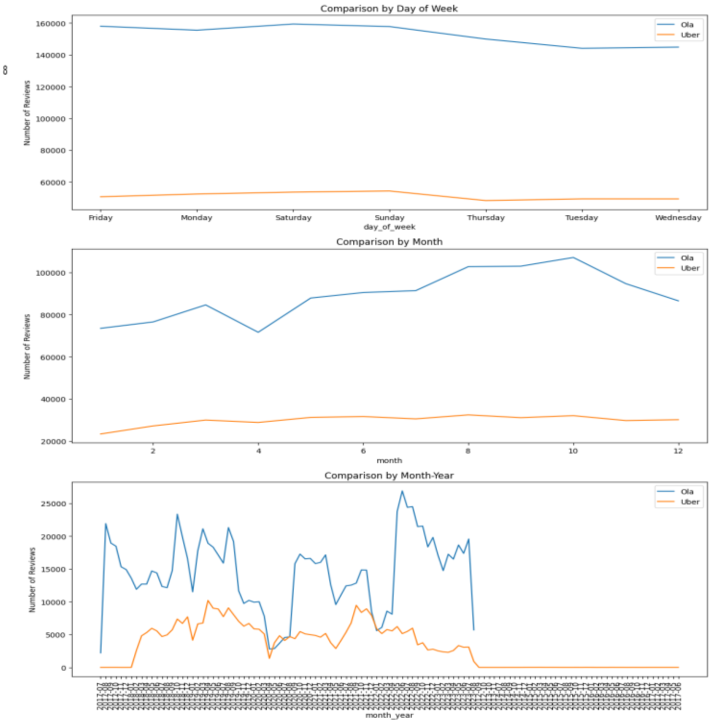

Figure-6: Comparision of data frame

6.1 Discussion:- In a Python-based project analyzing mobility services through machine learning, the outcomes would encompass insights into diverse facets of mobility, including user behavior, demand dynamics, route enhancement, and service efficacy. These potential findings could be dissected into several categories: firstly, scrutinizing user behavior to discern preferences across different transportation modes, peak usage periods, and common travel routes. Next, forecasting demand trends through machine learning models using historical data alongside external factors like weather or events, aiding providers in resource allocation. Additionally, optimizing transportation routes to minimize travel time, fuel consumption, and environmental impact would be a crucial aspect. Evaluating service efficiency metrics such as wait times, travel durations, and customer satisfaction rates would offer insights into service performance. Detecting anomalies within the data could uncover irregularities signaling issues like service disruptions or safety concerns. Moreover, crafting recommendation systems tailored to individual user preferences and context could enhance user experiences. Lastly, conducting environmental impact assessments of various transportation options would highlight opportunities for sustainability enhancements. Ultimately, this analysis aims to extract actionable insights to enhance mobility services and elevate user satisfaction levels.

7. REFRENCES

[1] Y. Indulkar and A. Patil, "Sentiment analysis of uber & ola using deep learning," IEEE, 2020.

[2] Y. Indulkar and A. Patil , "Comparative study of machine learning algorithms for twitter sentiment analysis,"IEEE,2021.

[3] S. Kumar, A. K. Singh, S. Bhushan, P. Kumar, and A. Vashishtha, "A comparative study on sentiment analysis of uber and ola customer reviews based on machine learning approaches, "SPRINGER, 2022.

[4] R. Sathya, "Uber data analysis using neural networks," Indian Journal of Natural Sciences, 2021.

[5] A. P. Kumar, K. Aashritha, M. Johnwesley, and J. Narasimharao, "A novel approach to analyze uber data using machine learning," IJARST, 2023.

[6] P. Banerjee, B. Kumar, A. Singh, P. Ranjan, and K. Soni, "Predictive analysis of taxi fare using machine learning," IJSRCSEIT, 2020.

[7] K. Khandelwal, A. Sawarkar, and S. Hira, "A novel approach for fare prediction using machine learning techniques," International Journal of Next-GenerationComputing, 2021.

[8] P. Saha, S. Guhathakurata, S. Saha, A. Chakraborty, and J. S. Banerjee, "Application of machine learning in app-based cab booking system: A survey on indian scenario," SPRINGER, 2021.

[9] K. Mehta, A. Shah, and S. Patel, "Cab fare prediction using machine learning," SPRINGER, 2022.