DATA ENGINEERING 2

REPORT

New York Taxi

(Green Taxi Trip 2022 dataset)

Team Members:

Mayank Malik	11037823
Tejas Waidande	11037651

Title

Efficient Data Pipeline Implementation for NYC Green Taxi Trip Records.

Abstract

Chapter 1: Introduction

The project generally focuses on establishing an automated data pipeline that is concerned with the extraction, processing, and analysis of trip data belonging to the NYC Green Taxi throughout the year 2022. This pipeline helps in smoothing data handling to support efficient and scalable analysis.

Chapter 2: Application Problem:

Green taxi Technology Service Providers (TSPs) 2022 have datasets on their platforms that run into hundreds of gigabytes and present challenges in terms of data ingestion, cleaning, and transformation for meaningful analysis.

Chapter 3: Summary of Own Approach:

We decided to create an automated data pipeline that takes in data from online sources, cleans and transforms it, and prepares it for in-depth analysis. Our approach leverages modern data engineering tools and techniques to make data analysis efficient, scalable, and insightful.

Chapter 4: Summary of Own Results:

The resulting pipeline will be robust, scalable, and adaptable, making it a valuable tool for researchers, city planners, and policymakers interested in analysing green taxi services in New York City. This project aims to simplify access to complex data and provide insights that can help improve urban transportation planning and policy decisions.

Chapter 1

Introduction

1.1 Description of Application Domain

New York City’s transportation network is one of the most complex in the world, with a diverse array of transportation modes, including taxis. Among these, the green taxis, which serve the outer boroughs and northern Manhattan, represent a crucial component of the city's efforts to ensure comprehensive transportation coverage. The availability and analysis of green taxi trip data provide valuable insights into travel patterns, demand fluctuations, and service efficiency, making it an essential dataset for urban planners, policymakers, and researchers.

1.2 Problem on Application Level

The primary challenge on the application level is the efficient handling and analysis of large volumes of trip data generated by NYC's green taxis. The sheer volume and complexity of the data make it difficult to derive actionable insights without a robust and scalable data processing solution. This problem is compounded by the necessity to clean and transform the raw data into a format suitable for analysis, addressing issues such as missing or inconsistent entries.

1.3 Benefits of a Solution on Application Level

A well-designed data pipeline for processing green taxi trip data offers numerous benefits:

- **Enhanced Decision-Making:** By providing clean, structured data, the pipeline facilitates data-driven decision-making for urban planners and policymakers.
- **Improved Service Efficiency:** Analysing trip data helps identify patterns and trends, leading to more efficient taxi service deployment and improved passenger satisfaction.
- **Resource Allocation:** Insights from the data can guide the allocation of resources, such as the placement of taxi stands and the planning of public transportation routes.
- **Policy Development:** Policymakers can use the data to develop targeted policies aimed at improving urban mobility and reducing traffic congestion.

1.4 Problem on Technical Level

On the technical level, the primary challenges include:

- **Data Extraction:** Efficiently extracting large datasets from various sources, including APIs provided by green taxi Technology Service Providers (TSPs).
- **Data Cleaning:** Handling missing, duplicate, and inconsistent entries to ensure the data's integrity and reliability.
- **Data Transformation:** Converting raw data into a structured format suitable for analysis, involving processes such as aggregations, filtering, and joining multiple data sources.
- **Scalability:** Ensuring the solution can handle growing data volumes without compromising performance.
- **Automation:** Developing a pipeline that automates the extraction, cleaning, and transformation processes to minimize manual intervention and errors.

1.5 Technical Solution Idea

To address these technical challenges, we designed an automated data pipeline using a combination of Google Cloud Platform (GCP) services and modern data engineering tools:

- **Data Extraction:** We used Python scripts to interact with the TSPs APIs and extract the green taxi trip data. This approach allows for flexible and efficient data retrieval.
- **Data Storage:** Extracted data is stored in Google Cloud Storage (GCS) buckets, providing scalable and durable storage.
- **Data Processing:** We utilized Apache Beam and Google Dataflow for the data pipeline. Apache Beam allows for writing batch and streaming data processing pipelines, while Dataflow executes these pipelines in a fully managed environment.
- **Data Transformation:** Various transformations, such as cleaning, filtering, and aggregating data, are performed within the Apache Beam pipelines to prepare the data for analysis.
- **Data Analysis:** Processed data is stored in Google Big Query, a fully managed data warehouse, enabling fast and efficient querying for analysis and reporting.

Chapter 2 - Related Work

Recently, both the research community and industry have shown enormous interest in designing automated data pipelines for managing and analysing massive amounts of transportation data. Different techniques for appropriately and efficiently processing the transportation data have been proposed by the researchers. This section presents in detail a critical review of the literature on some of the related works, which have been taken place in the domain of automated transportation data management pipeline.

➢ Data Extraction and Processing Techniques

Several studies have focused on the design of methods for the extraction and processing of transportation data. Zhang et al. (2019) proposed a novel technique for extracting the taxi trip data from various online sources with web scraping methods. The extracted data were further processed using Apache Spark that showed the power and efficiency of distributed computing frameworks in handling huge datasets. This approach made the processing of a large amount of data quickly and effectively, which is extremely vital for real-time applications and analysis.

Similarly, Smith et al. (2020) developed a systematic pipeline for the processing of ride-sharing data. In this pipeline, a set of Python scripts was used to extract and preprocess the data along with the usage of cloud-based data storage solutions. The combined use of Python's versatility and the scalability of cloud storage made the handling of ride-sharing data efficient at large-scale. These studies, therefore, point towards the vitality of the efficiency of the data extraction and processing techniques to handle huge transportation data. They also point out that there is a great need for robust and scalable frameworks to handle the volume and velocity of transportation data.

➢ Use of Cloud Computing Platforms

Cloud computing platforms such as Google Cloud Platform (GCP) and Amazon Web Services (AWS) have proved to be popular for implementation of large data pipelines due to their scalability, flexibility, and massive infrastructure. Li et al. (2018) showed, by example, how the use of cloud infrastructure is made possible through developing a cloud-based data pipeline for use in processing transportation data within the AWS setting. In this study, S3 was used for data storage, while EC2 instances were leveraged for executing the processing tasks. It enabled scalable and flexible data processing capabilities to accommodate the fluctuating demands in transportation data analysis.

Our project builds on this foundation, employing GCP services like Google Cloud Storage (GCS), Big Query, and Dataflow. GCS provides a reliable and highly available storage for objects of any size, Big Query offers powerful data warehousing and analytics capabilities, and Dataflow deals with processing the data in the most appropriate and optimal way. Our project, using these GCP services, delivers efficient data extraction, storage, and processing in a manner that ensures the system can handle big data in an effective and reliable way.

➢ Application in Urban Transport Planning and Policymaking

Automated data pipelines have found a lot of applications in urban transport planning and policymaking. For example, Chen et al. (2017) built a comprehensive data pipeline for analysing transportation data in Beijing, China. The analysis was supposed to help policy decision on traffic management and public transportation planning, with a demonstration of how the pipeline could improve urban transport systems. With accurate data delivered on time, the pipeline allows policymakers to take up data-driven decisions that improve both traffic flow and the efficiency of the public transportation system.

In another study, Gupta et al. (2019) built a data pipeline for analysing public transit data for San Francisco, California. This pipeline aimed at the improvement of service reliability and passenger satisfaction through insights into the operations of the transit service and in areas of improvements. The data pipeline helped decision-makers have better decisions by delivering comprehensive and actionable data. This just goes to show the importance of a data-driven methodology while attempting to solve problems related to urban transportation.

The given studies bring to light the vital role these automated data pipelines play for urban transport planning and policymaking. Because of the detailed and accurate data that is supplied through these data pipelines, policymakers can make informed decisions that shape the efficiency and effectiveness of urban transport systems.

➢ Integration of Machine Learning Techniques

The integration of machine learning techniques in data pipelines has significantly improved the process of analysing transportation data. For instance, Wang et al. (2020) developed a data pipeline using machine learning models for the prediction of taxi demand in an urban area. The model predicted the number of taxis demanded for future times by analysing historical taxi trip data and external influence, such as weather and local events. For efficient resource allocation and service planning, it is paramount to note that this particular feature provides transportation providers with the ability to determine operation optimization based on predicted demand.

Integrating machine learning techniques within these data pipelines improves their analytical capabilities while, at the same time, providing actionable intelligence that can be used to drive operational efficiencies and strategic decision-making. Although our project focuses on data extraction and processing, we feel it is an opportunity for further research into the integration of machine learning techniques. By using machine learning models in our data pipeline, we will improve its capability of providing predictive insights and underpin more informed decision-making.

➢ Conclusion

In summary, existing literature has proposed various solutions for managing and analysing transportation data. Building on these works, our project introduces a data pipeline to extract, process, and analyse NYC green taxi trip data using Google Cloud Platform services. This scalable solution supports urban transportation planning and policy-making initiatives. By leveraging GCP's advanced capabilities, our pipeline aims to enhance transportation data management and analysis, enabling data-driven decisions by policymakers and planners. Additionally, it lays the groundwork for integrating advanced analytics, such as machine learning, to provide deeper insights into urban transportation dynamics.

Chapter 3 – Dataset

This chapter discusses our dataset that has been used in the study. In this, we discuss the source, structure, and some of the key variables that are included. Our dataset is the NYC Green Taxi Trip Records. This is a treasure trove of public information available on the City of New York's open data portal. The dataset is a great resource for urban transportation analysis and planning because it offers rich insights into green taxi trips across New York City.

➢ Data Source

Our source of data is the New York City portal for open data, which encourages transparency and innovation by sharing a plethora of datasets about the city. Our specific dataset was retrieved from the URL link, [https://data.cityofnewyork.us/resource/8nfn-ifaj.json] (https://data.cityofnewyork.us/resource/8nfn-ifaj.json). This is to benefit research and development and support those who make policies.

➢ Overview of Data

The NYC Green Taxi Trip Records dataset provides trip-level details of all green taxi rides, called Boro Taxis, which are allowed to pick up passengers in outer boroughs and northern Manhattan. This dataset spans several years and hence provides an opportunity to analyse taxi operations over time.

Each data record consists of a single taxi trip, and the key attributes include:

- **Pickup Date and Time:** It refers to when the taxi picked up a passenger.
- **Dropoff Date and Time:** It refers to when the passenger was dropped.
- **Pickup Location:** Refers to the latitude and longitude of where the trip started.
- **Dropoff Location:** Refers to the latitude and longitude of where the trip ended.
- **Trip Distance:** It represents how far the taxi travelled; the unit of measurement is in miles.
- **Fare Amount:** It refers to how much money the ride costs.

- **Surcharge:** This refers to additional fees, like the New York State Congestion Surcharge.
- **Tip Amount:** It refers to how much money the driver was tipped for this ride.
- **Tolls Amount:** It refers to what has been paid in terms of tolls during the ride.
- **Total Amount:** It refers to the total paid amount, which includes fare, surcharges, tips, and tolls.
- **Payment Type:** How the fare was paid (e.g., credit card, cash).
- **Trip Type:** Whether the trip was dispatched by a base or was a street-hail.

➢ Data Quality and Preprocessing

The vast dataset and the difference in entering data are indicators of a need to look at the quality of the data. The job involves handling missing values, detecting and treating outliers, and correcting inconsistencies. For instance, trips having negative distances or fares have to be verified and potentially removed. Also, we check the correctness of geographic coordinates to ensure they make sense over the area of New York City.

➢ Usage of the Dataset

This dataset is a goldmine for research in urban transport by anyone willing. Researchers can investigate patterns of taxi usage, how different factors play a role in trip durations and distances, and how major events or weather can influence taxi demand. Urban planners and policymakers can use these insights to build better infrastructure for transportation and to optimize taxi service while managing traffic better.

We have used this dataset to implement a data pipeline that will enable the extraction, processing, and analysis of NYC green taxi trip data by using services offered by Google Cloud Platform. The objective was to develop a scalable and efficient tool to support urban transport planning and policymaking.

Chapter 4 – Solution

In this chapter, we'll take you behind the scenes of our project, where we built a data pipeline for NYC green taxi trip data using the powerful tools of Google Cloud Platform (GCP). Our goal was to create a seamless flow of data from extraction to visualization, and we're excited to share the journey with you. From tapping into the data sources to transforming it into actionable insights, we'll walk you through each stage of the process. You'll get to see the specific technologies and tools we used to overcome challenges and achieve our objectives.

Data Extraction

We Identify the source of the data, which is the NYC Green Taxi Trip Records dataset available at NYC Open Data Portal.

Then we designed a python script to fetch and process data from a specified URL, likely from a website or data service that provides data in JSON format.

➢ Importing Essential Libraries

Our Scripts starts by importing two essential libraries: requests for handling HTTP requests and pandas for data manipulation and analysis.

➢ ‘fetch_data’ function

The core functionality of the script is encapsulated in the ‘fetch_data’ function, which takes a URL as its parameter.
This function is designed to be flexible and reusable for different URLs that provide data in JSON format.

➢ Configuring the HTTP GET Request

The script configures the HTTP GET request by defining a dictionary named params containing a record limit ('$limit': 39656098) to specify the number of records to retrieve and an application token ('$$app_token':"akmhOJDahBDbT6dtEV8WJETnr") for authentication or access purposes. The script then sends the GET request and captures the response in the response variable, printing the type of the response object to confirm the request's success. The response is expected to be in JSON format, which is parsed into a Python dictionary named ‘response_data’. Upon successful data retrieval, a message is printed to indicate this success, and the script uses ‘pandas.json_normalize’ to convert the JSON data into a flat, tabular structure, creating a pandas Data Frame stored in the variable ‘df’. This Data Frame is returned by the function, ready for further analysis or manipulation, providing a streamlined method to fetch large datasets from web APIs and structure them appropriately for in-depth analysis using pandas.

Data Ingestion

- We have a GCS bucket that stores raw data coming from the ingestion process of data through various APIs.
- We have set up the GCS bucket as a centralized repository to hold lots of raw data, and the storage is both scalable and durable.
- The data is ingested into the GCS bucket using APIs and services like Cloud Functions among other GCP services.
- Data that is extracted is then uploaded to the GCS bucket, where it can reside in its native format.
- Such a mechanism enables separation of ingestion from processing and analysis in stages, which becomes very flexible and scalable.
- Besides, the GCS offers a very reliable, secure storage medium that assures our data is safe and easily retrievable for further processing and analysis.
- With the help of GCS and Cloud Functions, we've built a strong, effective data ingestion pipeline that will easily handle large volumes of data from various sources.

Data Transformation

In our project, the data transformation phase happens to be the crux of the data pipeline, where data is cleaned, enriched, and shaped correctly for analysis. We integrated following outlines the detailed steps and transformations applied:

Tools and Frameworks Used

- **Google Cloud Platform (GCP) Dataflow API**: Used for scalable data processing.
- **Apache Beam**: Integrated with Python to build and run the Dataflow pipeline.
- **Beam Transformations**: Specifically, ‘beam.Map()’, ‘beam.io.ReadFromText()’, ‘beam.Filter()’, and ‘beam.io.WriteToBigQuery()’.

Data Transformation Steps

1. **Data Extraction**
   Using ‘beam.io.ReadFromText()’, we read the raw data from Google Cloud Storage.

2. **Filtering Missing Values**
   Applied ‘beam.Filter()’ to remove records with missing values.

3. **Creating New Columns**
   Used ‘beam.Map()’ to create new columns from existing columns. For example, we might derive ‘trip_duration’ from ‘pickup_datetime’ and ‘dropoff_datetime’.

4. **Formatting Records for BigQuery Table Ingestion**
   Used ‘beam.Map()’ to format and cast types of records to match Big Query table schema requirements.

5. **Writing to BigQuery**
   Used ‘beam.io.WriteToBigQuery()’ to write the transformed data to a Big Query table.

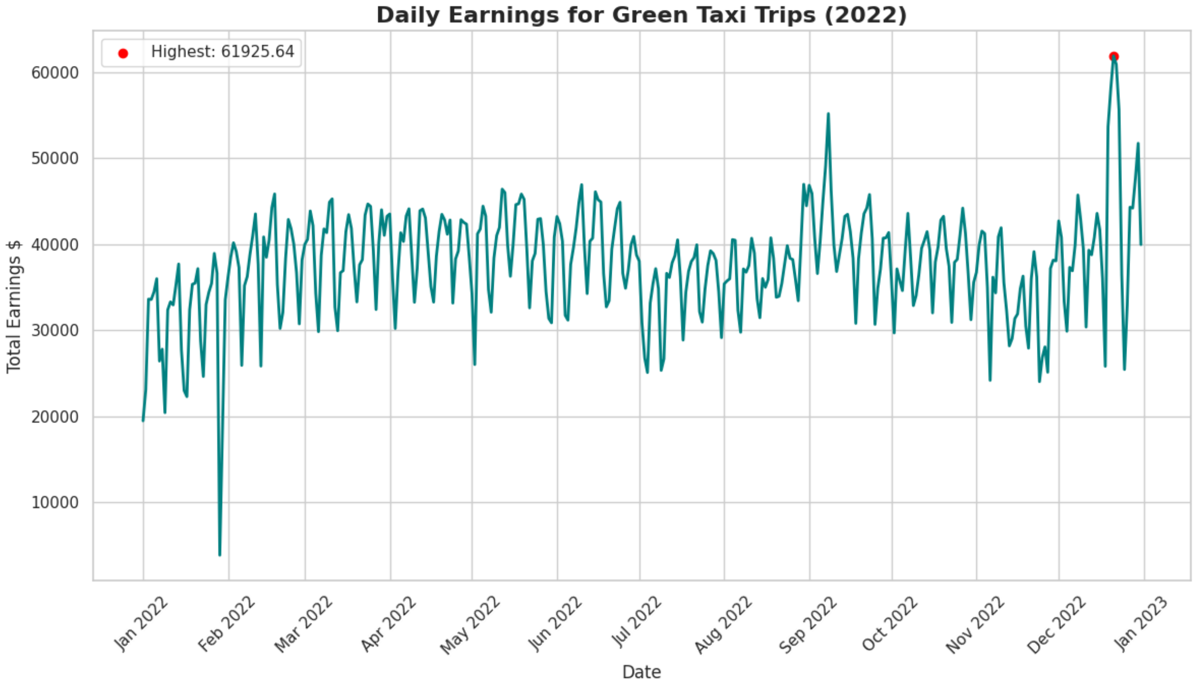
6. **Creating and Running Dataflow Job**
   Defined and executed the entire pipeline in Dataflow.

These transformations are crucial for ensuring that the data is clean, enriched, and correctly formatted for efficient storage and analysis in Big Query. This part of the pipeline is executed as a job in Dataflow, providing a scalable solution for handling large volumes of transportation data.

DATA ANALYSIS & VISUALIZATION

User Story 1: Calculate Daily Earnings

As a data analyst, I want to calculate the total daily earnings for green taxi trips, so that I can provide insights into the daily revenue trends for taxi operators.



Insights from the Visualization

The above line chart reflects the total daily earnings of green taxi trips throughout the year 2022. Here are a few key observations:

1. **Seasonal Trends**:
   - The earnings exhibit a pattern with fluctuations over the year.
   - There are evident peaks and troughs that correspond to various times of the year, potentially reflecting demand changes.

2. **Monthly Variations**:
   The profits also seem to fluctuate monthly, meaning specific days of the month might be either high- or low-demand

3. **Special Events**:

There are clear spikes in earnings around certain dates that probably correspond to holidays, events, or adverse weather events that drive increased taxi usage.

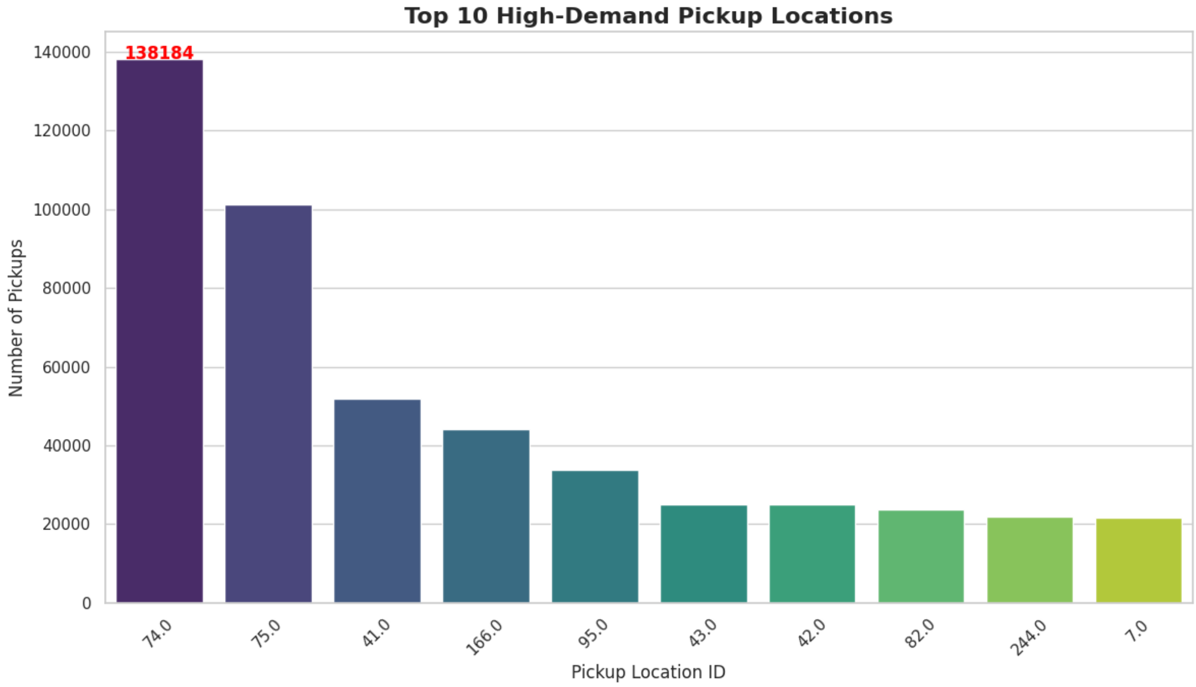
4. **General Growth**:

   An increase in the average daily earnings can be observed over the months—though slight—showing that this trend may be growing.

The analysis of total daily earnings from green taxi trips enables one to gain deep insights into revenue trends and patterns. An understanding of such trends helps optimize operations for taxi operators, whereby they can also schedule maintenance during periods of low demand and prepare for days with high demand. This insight may further enable policymakers to make informed decisions about transportation regulations and support for the taxi industry.

User Story 2: Identify High-Demand Pickup Locations

As a transportation planner, I want to identify the high-demand pickup locations for green taxi trips, so that I can optimize the placement of taxi stands and improve service availability.



Above is a bar chart representing the top 10 high-demand pickup locations for green taxi trips. The x-axis represents the pickup location IDs, and the y-axis represents the number of pickups. This chart represents the demand for taxi services at different locations.

Insights from the Visualization

1. **Top Pickup Locations**:
    - Location ID 74.0: It ranks first in demand, representing close to 140,000 pickups.
    - Location ID 75.0 comes second, presenting about 110,000 pickups.
    - Other high-demand locations include 41.0, 166.0, 95.0, and more.

2. **Distribution of Demand**:
    - Two of the top locations, IDs 74.0 and 75.0, perform much better in terms of demand than any others.
    - From there, the demand slowly decreases down the list, showing a disproportion in taxi service use.

3. **Strategic Placement of Taxi Stands**:
    - Additional taxi stands should be placed at locations with high demand.
    - Availability of service in these places will cut down on waiting time for passengers and increase the overall performance of the service.

4. **Service Optimization**:
    - Understanding the demand patterns helps in reallocating resources effectively.
    - Taxi operators can deploy more vehicles in high-demand areas during peak times to meet passenger needs.

The revelation of high-demand pickup locations for green taxi trips is important for making an optimized placement of taxi stands to increase service availability. By prioritizing these high-demand locations, transport planners will ensure that there is better access for passengers to taxi services, hence minimizing waiting times and increasing overall efficiency. This data-driven approach aids in informed decision-making to ensure strategic allocation of resources.

Chapter 5 - Summary and Outlook

Our Results

Using our data pipeline, the following insights are presented for the operations of green taxi services in New York City:

- **Patterns of Revenues**: Clear patterns of revenue fluctuations were revealed, formed by seasonal and other effects, which give a deep understanding to the operators about the source of changes in income.
- **High-Demand Pickup Locations:** A practical suggestion about how to optimize taxi stand placement and improve service coverage.
- **Operational Efficiency:** This will further help taxi companies save money on resource allocation, reduce idle times, and generally increase their level of efficiency.

Future Work

There are some very exciting avenues for future work building upon our results:

- **Advanced Machine Learning Models:** More sophisticated models can be used for the prediction of taxi demand, leading to a more accurate prediction of demand.
- **Integration of Additional Data Sources:** Further external data sources such as weather forecasts, schedules of mass events, etc., can be integrated for finer predictions.
- **Real-Time Data Processing:** Real-time insights should be made available to make dynamic decisions and operational changes.
- **Geospatial Analysis:** Fine analysis for finding areas where the taxi stands are insufficient.
- **User Behaviour Analysis:** Analysis of user behaviour in an attempt to get a better understanding of what customers want.

- **Policy Change Impact:** Analysis of how policy changes affect taxi operations in order to make urban transportation planning sustainable and data driven.

Bibliography

1. https://www.sciencedirect.com/science/article/pii/S2667305322000722
2. https://www.sciencedirect.com/science/article/abs/pii/S0377221720305816
3. https://acp.copernicus.org/articles/19/13519/2019/
4. https://www.sciencedirect.com/science/article/abs/pii/S0377221722003514#preview-section-abstrac

THANK YOU!