Analyzing NYC Subway Turnstile Data: Understanding subway ridership pattern and Commuter Behavior

Avinash Vijayarangan av3134[a], Nagharjun Mathimariappan nm4074 [a], Nishal Sundarraman ns5429[a] and Vikram Balaji vb2284[a]

[a]New York University

Abstract. The project titled "Analyzing NYC Subway Turnstile Data: Understanding subway ridership pattern and Commuter Behavior" aims to address the problems of regulating traffic during rush hours and maximizing ad viewership at subway stations. By analyzing MTA turnstile data for the past 10 years and performing exploratory data analysis using big data tools such as Pyspark, Spark ML, Kafka, and MongoDB, this project seeks to infer insights into subway ridership patterns and commuter behavior. The findings of this analysis will enable the identification of stations with the highest foot traffic and the prediction of which stations are likely to experience huge foot traffic at specific hours. This report highlights the importance of leveraging big data tools and analysis to improve the management of essential public infrastructure such as the New York City subway system.

1 Why is this a Big Data Problem?

The MTA dataset contains over 11 million records spanning 10 years (2014-2023), encompassing information about each turnstile's cumulative number of entries and exits for every four-hour period for 114 stations across New York City. However, processing such a large dataset on a single machine would take a considerable amount of time. As such, it is more practical to scale the data processing using big data techniques horizontally.

Horizontal scaling is more cost-efficient than vertical scaling, as it allows for multiple machines to work in coordination with accuracy. This approach can significantly reduce the time taken to process the data. This is where PySpark comes in. It is an incredibly useful tool that can preprocess data and perform necessary actions on it. Additionally, Spark ML can efficiently train and test machine learning models on preprocessed data.

To stream the data and create a pipeline for processing, Kafka is an ideal solution. By utilizing these big data techniques, we can create a cost-efficient, accurate, and timely solution for analyzing the MTA dataset. This will enable us to identify stations with high foot traffic and predict foot traffic at specific hours for Foot Traffic prediction purposes.

Overall, by horizontally scaling the data processing, we can reduce the time taken to process the data, while maintaining accuracy and cost efficiency. This will allow us to make effective use of the large dataset provided by the MTA and extract valuable insights that can aid in improving the management of public infrastructure in New York City.

2 Dataset

The MTA dataset consists of over 11 million records for the past 10 years and the following are the features of the dataset:

1. Dataset: http://web.mta.info/developers/turnstile.html

2. Size: 12GB

3. Schema: C/A, UNIT, SCP, STATION, LINENAME, DIVISION, DATE, TIME, DESC, ENTRIES, EXITS

4. Description:
   C/A = Control Area (A002)
   UNIT = Remote Unit for a station (R051)
   SCP = Subunit Channel Position represents a specific address for a device (02-00-00)
   STATION = Represents the station name the device is located at
   LINENAME = Represents all train lines that can be boarded at this station
   DATE = Represents the date (MM-DD-YY)
   TIME = Represents the time (hh:mm: ss) for a scheduled audit event
   ENTRIES = The cumulative entry register value for a device
   EXITS = The cumulative exit register value for a device

3 Architecture

The MTA Dataset was critical for our project, but it was available only in a non-downloadable text format. We overcame this issue by writing a Python script that scraped the dataset of the past 10 years onto our local disk. The dataset contained over 11 million records for the past 10 years from 2014 - 2023, and each record provided information about each turnstile's cumulative number of entries and exits for a 4-hour period for all the 114 stations that cover the entirety of NYC.

We used PySpark to upload the records onto a PySpark Dataframe and cleaned it, void of null values and data discrepancies, and

created the necessary columns required to calculate the net foot traffic at a given station, a given turnstile at any given time. We also performed EDA on the pre-processed data using Pandas and PySpark functions, which helped us gather valuable insights into the data. As a result of this EDA, we identified the top 5 most populated stations for the past 10 years.

After performing the EDA, we used Spark ML to train five different regression models, each corresponding to one of the five stations under observation. The models were trained to predict the Foot Traffic for these 5 stations at any said time for the first 8 years of the dataset, from 2014-2021. We stored the coefficients of these models in a collection in MongoDB to be used later for testing the data.

In a separate pipeline, we streamed the data for the years 2022 and 2023 through Kafka. These data were later plugged into the pre-trained models, and the coefficients stored in MongoDB were retrieved to produce near-accurate predictions of the Foot Traffic in the said 5 stations for any given time and a turnstile. By scaling horizontally and using Big Data tools such as PySpark, Spark ML, Kafka, and MongoDB, we were able to process the massive dataset efficiently, making accurate predictions and valuable insights that can help regulate traffic during rush hours and optimize ads at these stations with high viewership.

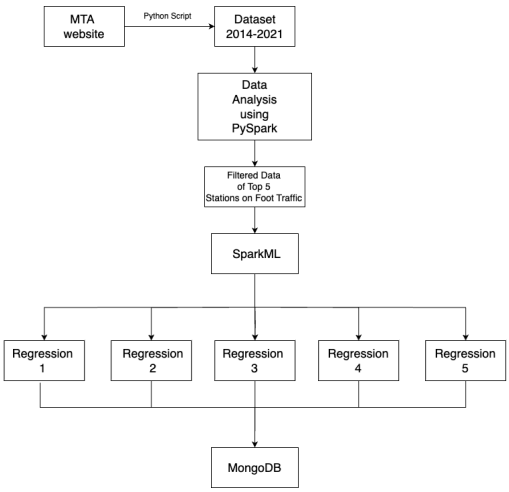

Figure 1. Architecture

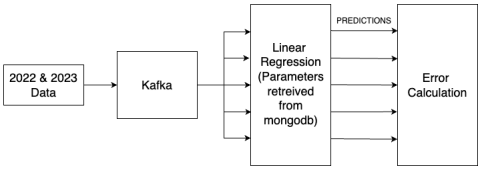

Figure 2. Architecture

4 Exploratory Data Analysis

EDA (Exploratory Data Analysis) is a crucial step in analyzing and understanding complex datasets. In this section, we will discuss the EDA conducted for analyzing NYC Subway Turnstile Data, focusing on subway ridership patterns and commuter behavior.

The goal of this EDA was to gain insights into the busiest subway stations, average foot traffic by year and month, the impact of holidays on ridership, and the relationship between ridership and seasons. By examining these aspects, we aimed to uncover valuable information that can assist in optimizing subway operations and enhancing commuter experiences.

1. Busiest subway stations in NYC
   - Identified the top 10 busiest subway stations in NYC based on average ridership per day.
   - We analyzed the data and found that the 5 busiest stations are 34 ST Penn, 34 ST Herald SQ, Grand Central-42 ST, 14 ST Union SQ, 86 ST.

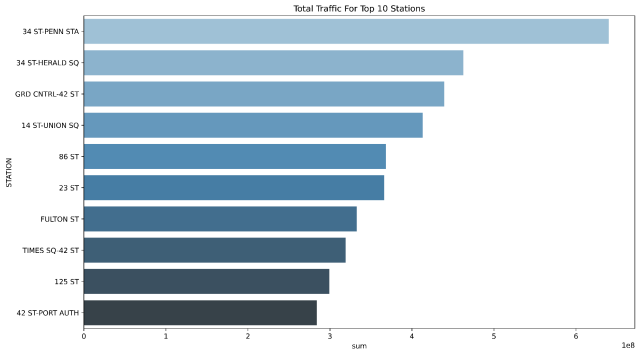

Figure 3. Total Traffic for top 10 stations

2. Foot Traffic
   - Average foot traffic in the NYC subway system was examined by year.
   - The COVID-19 pandemic had a profound impact, causing a significant decrease in subway ridership in 2020 and 2021 as seen in Fig 4.
   - Factors such as remote work, school closures, and reduced tourism contributed to the decline in ridership.
3. Average daily entries for Holiday vs Non-holiday
   - Average daily entries were analyzed for holidays and non-holidays in the NYC Subway Turnstile Data.
   - From Fig 5, it can be revealed a decrease in subway ridership during holidays compared to non-holiday periods.
   - Specifically, Christmas showed the lowest average daily entries compared to any other day of the year.

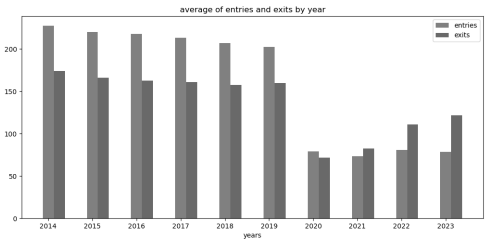

Figure 4. Average Foot Trafffic by year

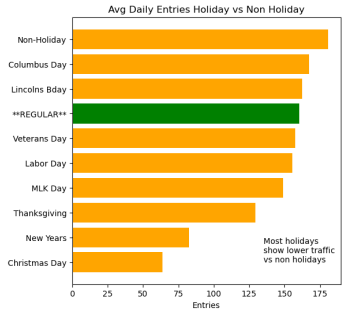

Figure 5. Average daily entries for Holiday vs Non-holiday

5 Results

The successful training of the regression models is a significant achievement as it enables accurate predictions of the foot traffic in the 5 stations under observation. The coefficients of these trained models were stored successfully in MongoDB, ensuring their accessibility for future use. To measure the accuracy of the models, the Root Mean Square Error (RMSE) was used as the error metric during training. It was found that the 5 trained models had RMSE values as follows:

34th Herald Square	310.184
Penn Station	273.345
Grand Central	308.988
14th Street	344.434
23rd Street	275.161

Figure 6. RMS Error for the 5 regression models

6 Conclusion

In conclusion, our exploratory data analysis (EDA) of the NYC Subway Turnstile Data has provided valuable patterns and insights, with a particular focus on the busiest stations. Through our analysis, we have identified significant findings that shed light on subway ridership patterns and commuter behavior.

One noteworthy observation is the consistent popularity of certain stations, which experience high levels of foot traffic on a daily basis. This knowledge can assist in optimizing operations and resource allocation within the subway system to ensure efficient transportation services.

Moreover, our analysis revealed the advantages of streaming data through Kafka compared to storing it solely in a PySpark dataframe. Streaming data through Kafka allows for real-time processing and analysis, enabling prompt insights and the potential for immediate action based on emerging trends or anomalies in subway ridership.

Overall, our EDA has provided a comprehensive understanding of subway ridership patterns and commuter behavior in NYC. These insights can inform decision-making processes, enabling improvements in subway operations, resource allocation, and commuter experiences. By leveraging advanced technologies like Kafka and Spark ML, we can harness the power of real-time data analysis and scalable machine learning to further enhance the efficiency and effectiveness of the subway system.

7 Limitations and Future Works

The turnstile entry and exit data can also help identify the number of individuals who are using the subway network without paying, by calculating the difference between the number of people who have exited and entered. This information can be utilized to determine the potential profits of implementing a more secure turnstile system.

By utilizing the current foot-traffic data, it is possible to establish the number of individuals entering and exiting subway stations, which can be leveraged to implement variable pricing for advertising across different subway stations. For example, the advertising cost at Penn Station can be set higher than other subway stations due to its higher visibility and footfall.

To improve the accuracy of the model, we can consider incorporating weather data to obtain better correlations between the features.

Acknowledgements

We would like to express our sincere gratitude to Professor Juan Rodriguez for his invaluable comments in significantly improving this report.

References

1. https://github.com/Nagharjun17/BigDataProject