Anushka Tawte (at5849)
Nithinaakash Sivaprakash (ns5840)
Neel Gandhi (njg9191)

Crash NO! MO!

Introduction

Urban environments face significant challenges from traffic accidents, impacting public safety and imposing substantial economic costs. As cities continue to grow, the integration of advanced data analytics becomes essential for improving traffic management and safety. This project leverages detailed traffic and environmental data to develop a predictive model aimed at identifying and mitigating traffic accident hotspots in urban areas.

Motivation

The driving force behind this project is the critical need to enhance traffic safety and reduce the economic burdens associated with accidents in urban settings. Utilizing the wealth of data available from sources like NYC open traffic data, this initiative aims to harness insights on traffic conditions, including flow, volume, and speed, along with weather and urban infrastructure data. The objective is to develop a predictive model that identifies potential crash hotspots, enabling proactive measures to prevent accidents and ultimately save lives while fostering more efficient urban environments. This approach not only supports immediate traffic management decisions but also informs long-term planning and policy-making to improve road safety

Data Sources and the Dataset

There were multiple datasets that were utilized for this particular project. This ensured that all the important and contributing factors were considered giving a more accurate and structured outcome. The datasets and their uses are described below.

Historical Crash Data:
Analyzed types of crashes, including the nature and severity of each incident. Calculated accident severity using a function that accounted for the number of injuries, fatalities, contributing factors, and vehicles involved. This severity metric is crucial for assessing and predicting the impact of crashes in different areas of the city.
Link: Historical Crash Data

Road Network Data:
We transformed the road network information from its original NetworkX format into a usable DataFrame format to facilitate analysis. Additionally, we developed a function that calculates the number of road intersections within a specified polygonal area. This feature is particularly valuable, as our preliminary findings indicated a high frequency of accidents at intersections, especially at T-intersections.
Link: Road Network Data

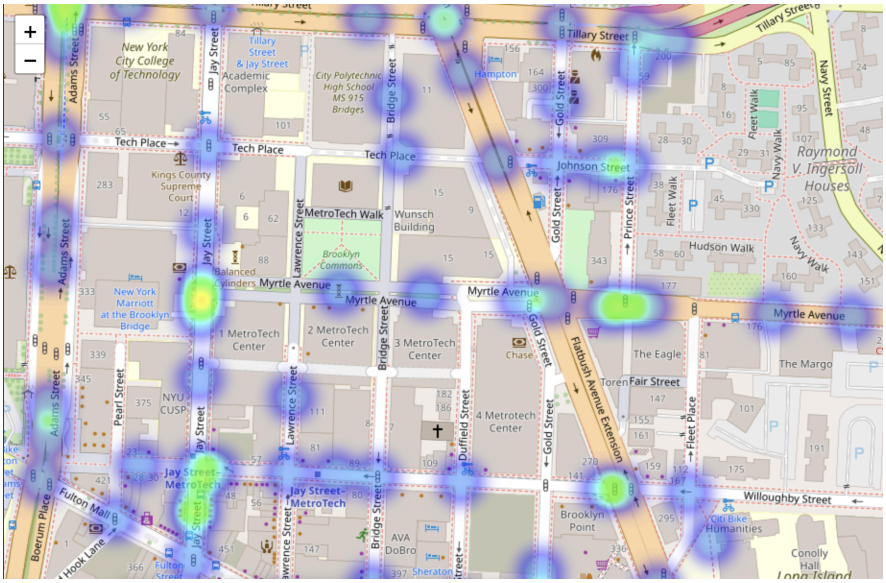
Weather Data (via API):
Retrieved key weather metrics such as maximum, minimum, and mean temperatures for specific locations. Extracted additional weather conditions including precipitation, snow, and rain, integrating these into our dataset to enhance the model's ability to factor in weather-related variables, which are often significant contributors to accident rates.

Infrastructural Data:
Included data on key infrastructural elements like nearby schools and parks. This data helps refine our understanding of traffic patterns and potential pedestrian-heavy areas, which are essential for predicting accident hotspots.
Data: Schools, Parks

Traffic Data:
Utilized data detailing traffic flow, density, and speeds to gain insights into typical driving conditions across different times and areas. This information is vital for understanding the dynamics that contribute to traffic accidents.
Link: Traffic Data

Methodology

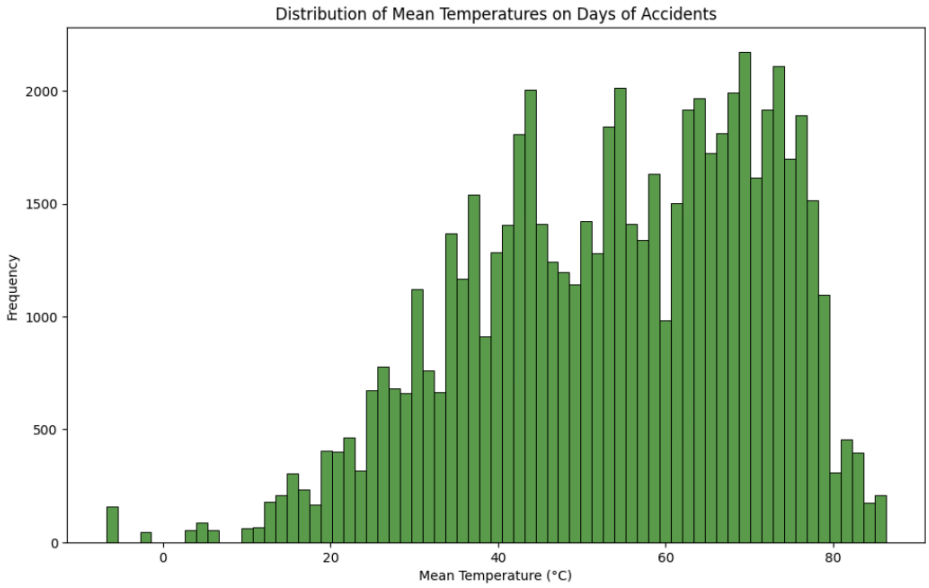
Data Cleaning and Analysis: Our project began with a comprehensive data cleaning and initial analysis of the NYC crash dataset. This foundational phase was crucial to ensure data quality and usability for further exploration. In this phase we realized that historically intersections had accidents. Which led us to add some kind of road network data to our final dataset.



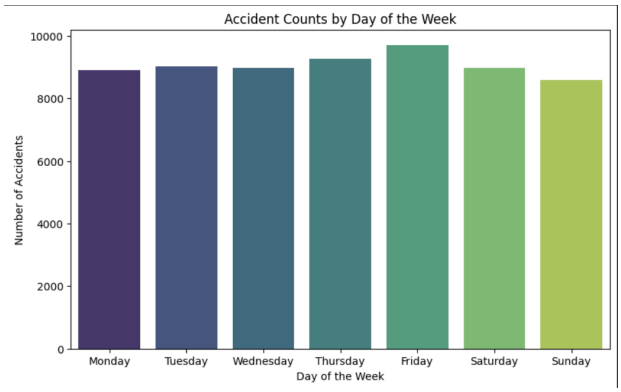
Historical Crash sites Plotted on NYC Map

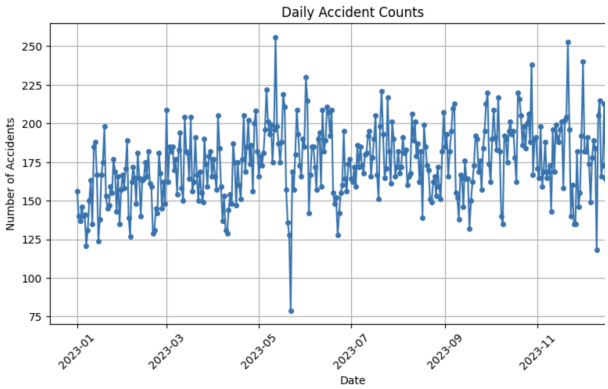
To augment our dataset, we integrated the NYC road network data, formatted in NetworkX. This integration enriched our dataset significantly, enabling more precise predictions about accident locations

Analysis of weather conditions revealed that most accidents occurred under normal temperature conditions, with a decrease during extreme weather events, suggesting less driving activity in adverse conditions.

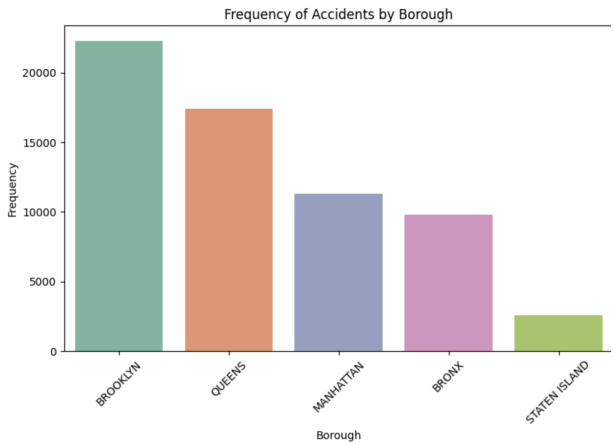


Our temporal analysis indicated that Fridays have a slightly higher incidence of crashes, likely due to increased social activities at the week's end. A seasonal trend was also observed, with May, and to an extent, November and December, experiencing higher fluctuations in accident rates, likely influenced by increased tourist activity (Figure below).





Spatial analysis showed that Brooklyn recorded the highest number of accidents, followed by Queens and Manhattan. This trend correlates with the population distribution and is indicative of the relationship between population density and traffic accidents.



Name	Population Estimate 2022-07-01
Bronx	1,356,476
Brooklyn (Kings County)	2,561,225
Manhattan (New York County)	1,597,451
Queens	2,252,196
Staten Island (Richmond County)	490,687
New York City	**8,258,035**

These analyses provided valuable insights into the dataset and were instrumental in guiding our choice of predictive models. Through comprehensive graphing and additional analyses, we recognized the critical importance of visual and interactive representations of data. Consequently, we developed an interactive dashboard that enhances user engagement and understanding by allowing them to explore data dynamically. The significance and utility of these tools will be further discussed in the section on our proposed solution.

Project Overview

1. **Utilizing Apache Spark:**

    - **Data Integration:** To manage and analyze the vast volumes of disparate data efficiently, we employed Apache Spark, a powerful distributed data processing engine. Spark's ability to handle big data at scale was instrumental in merging datasets from various sources such as historical crash records, road network configurations, and traffic patterns into a unified data framework. This integration process is critical, as it ensures that all relevant variables are considered when modeling traffic accidents.

    - **Preliminary Data Processing**: Spark was also used for the initial stages of data cleaning and preparation. This included filtering out inconsistencies, dealing with missing values, and transforming data into formats suitable for further analysis. By performing these tasks in Spark, we leveraged its fast processing capabilities to speed up the data preparation phase, ensuring that data was ready for more detailed analysis and modeling efficiently.

2. **Advanced Data Manipulation with Dask:**

    - **Further Data Processing:** After the initial data integration with Spark, Dask was utilized for more complex data manipulations and to handle specific, compute-intensive tasks that required fine-grained control over data operations. Dask's dynamic task scheduling and parallel processing abilities made it ideal for processing large datasets that were too cumbersome for single-machine tools.

    - **Custom Data Operations:** Specific functions, such as calculating the number of road intersections within certain geographical boundaries or performing detailed time-series analyses on traffic flow data, were executed using Dask. This allowed for optimized performance and scalability in data manipulation tasks, which were crucial for generating the final dataset used in predictive modeling.

    - **Data Acquisition and Integration:** Utilizing Dask, our team managed the fetching and cleansing of an array of datasets, which included live and historical traffic data, historical weather conditions, complexities of the road network graphs, and, notably, historical crash data. This

comprehensive data collection aimed to lay a robust foundation for the subsequent modeling processes.

- **Efficient Data Merging Techniques:** Integrating crash data with meteorological data was streamlined due to the consistent presence of zip codes and date stamps in both datasets, allowing for straightforward merging. In contrast, the historical traffic data posed integration challenges due to missing zip codes and inaccurate geographical coordinates. Extensive processing in Dask corrected and extracted these values, ensuring precision in our analyses.

- **Geospatial Analysis for Enhanced Insights:** For each recorded crash, a geographical circle with a radius of 300 meters was defined around the crash site. Within this zone, we calculated the average traffic conditions at the precise time of the crash. This method provided a localized snapshot of traffic dynamics near crash sites, crucial for understanding the conditions leading to accidents. A similar approach was used to amalgamate road network intersection data with traffic and crash statistics, providing a multi-dimensional view of the factors influencing each crash event.

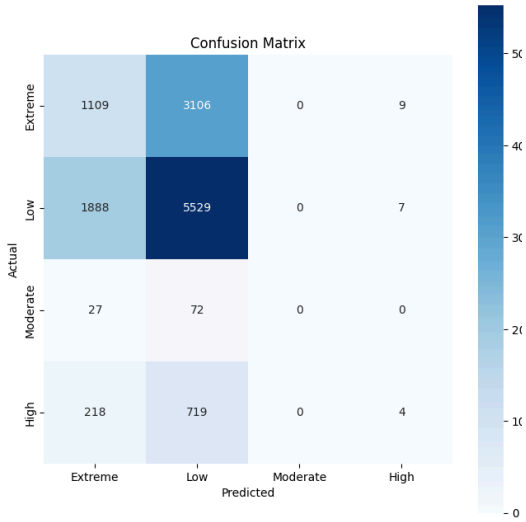
3. **Model Selection and Rationale**
   Given the complexity and size of the data, two models were selected for experimentation:

   - **K-Nearest Neighbors (KNN)**: Chosen for its ability to correlate various crash conditions, KNN provides a measure of similarity to high crash spots, which is pivotal in predicting traffic accidents. However, due to its sensitivity to the scale of data, and the vastness of our dataset, KNN struggled to perform optimally.
   - **Random Forest**: We selected this model for its robustness against overfitting and its efficacy in handling imbalanced data. Random Forest is also well-suited for understanding complex data structures, which makes it ideal for our diverse dataset composed of traffic, road network, and weather data.

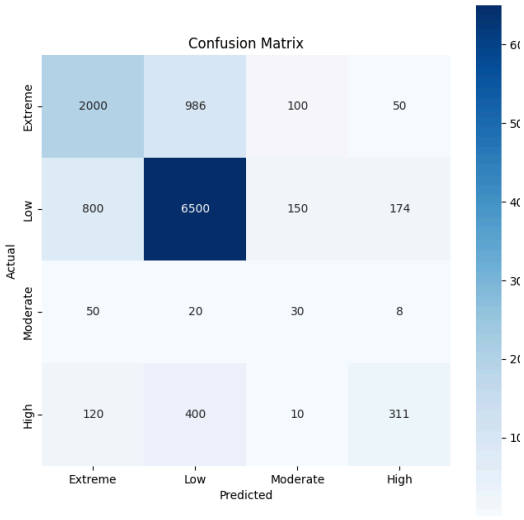
4. **Implementation and Performance**

To overcome the challenges of large data volumes, DaskML was utilized to facilitate distributed training, particularly for the Random Forest model. This approach significantly reduced training times and improved model scalability.

During the comparative analysis, Random Forest demonstrated superior performance over KNN. It was particularly effective in dealing with non-linear data and showcased a better understanding of the data complexities. The confusion matrix revealed that Random Forest could accurately predict rare events in the dataset—a crucial advantage where KNN failed to predict even a single correct value for those data points.



KNN



Random Forest

5. **Application Deployment:**
   The actionable insights generated by the predictive model are visualized through an interactive dashboard developed using Streamlit. This dashboard provides real-time updates on traffic conditions and accident probabilities, making it an essential tool for both the public and decision-makers. By offering real-time data visualization, the dashboard serves not only as a public information utility but also supports city planners and traffic management authorities in crafting and implementing effective safety measures tailored to specific urban conditions. This deployment effectively bridges the gap between predictive analytics and practical, on-the-ground application, enhancing urban traffic safety management.

Results

Streamlit: Serves as the frontend framework for creating the interactive dashboard.

1. **Technology Stack:**

    - **Pandas and Dask**: Used for efficient data handling and parallel computing to manage large datasets.
    - **Folium**: Provides mapping functionalities to visualize geographical data.
    - **Geopandas and Shapely**: Handle geospatial data operations.
    - **Matplotlib, Seaborn, and Altair**: Generate various static and interactive visualizations.
    - **Pyproj and Socrata**: Facilitate coordinate transformations and data fetching from external APIs.

    The application is designed with several interactive components and visualizations to enhance the user experience and provide detailed insights:

2. **Interactive Map Visualization:**

    Displays traffic accidents and congestion data using heatmaps.
    Allows users to adjust the map view to focus on specific areas, with the data updating dynamically based on the map bounds.

3. **Real-Time Data Processing:**

    Incorporates live traffic data to calculate average speeds and congestion levels within selected map areas.
    Fetches and processes weather data in real time based on the user's selection, influencing the accident risk predictions.

4. **Customizable Data Queries:**

    Users can toggle conditions like rain presence and high traffic to see their impact on traffic patterns and accident risks.
    The dashboard updates automatically to reflect changes in these conditions, showing how environmental factors might influence accident severity.

5. **Predictive Analytics:**

Utilizes machine learning models (KNN and Random Forest) to predict accident severity based on traffic, weather conditions, and time variables.
Displays predictions directly in the dashboard, providing immediate insights into potential traffic safety hotspots.

6. **Statistical Analysis Tools:**

   Offers detailed statistical views on accidents, including correlations between accident severity and environmental conditions like rain.
   Generates bar charts and cross-tabulations for an in-depth analysis of the data.

7. **Application Workflow**

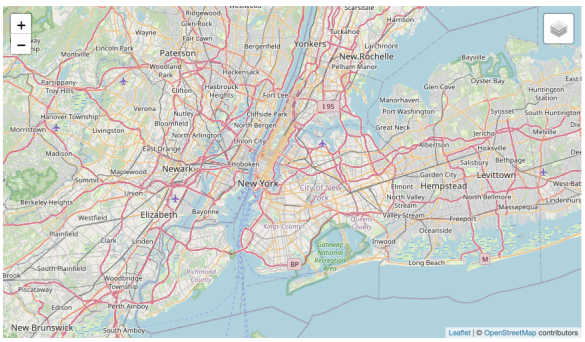
   - **Data Loading**: Imports traffic and weather data from various sources, including city APIs and local CSV files.
   - **User Interaction**: Users interact with the map to select specific areas or set conditions like weather and traffic.
   - **Data Processing**: Executes data transformations and calculations in real-time using Dask for parallel processing and efficient computation.
   - **Visualization**: Updates visualizations dynamically based on user inputs and data queries. Folium is used for mapping, while Altair handles other graphical plots.
   - **Prediction and Analysis:** Applies pre-trained ML models to predict accident severity and displays the results alongside statistical analyses.

8. **Challenges and Solutions:**

   - **Data Volume**: Handling large datasets was challenging, effectively addressed by integrating Dask for parallel data processing, ensuring responsiveness and speed.
   - **Real-Time Data Integration**: Fetching and integrating real-time data from APIs required careful error handling and caching strategies to optimize performance and reduce unnecessary API calls.

Deploy

Crash No Mo!



With Heavy Rain

With High Traffic

Crash Severity Prediction

Potential Hot Spot

Avg Traffic Speeds

32 MPH

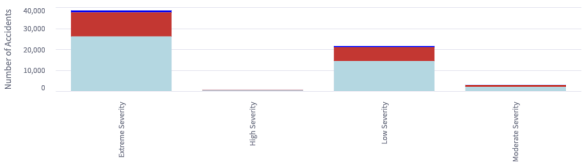
Weather

76.7 °F, Partially cloudy

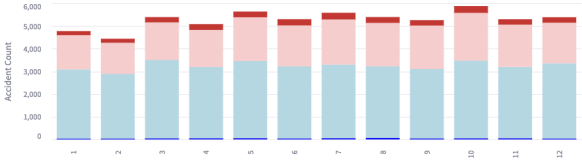
Intersections

345959

Rain vs. Accident Severity

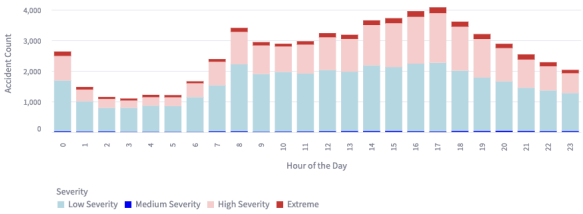


Accident Severity vs. Month

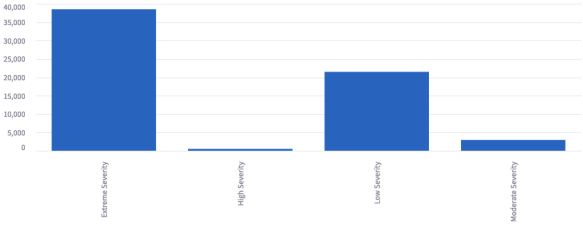


Deploy

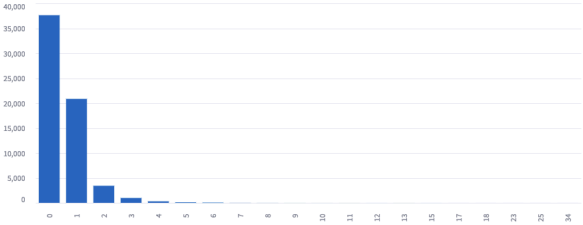
Accident Severity vs. Hour of Day



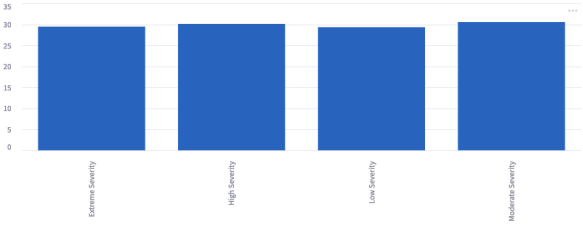
Accident Severity vs. Intersections



Number of Persons Injured Distribution



Speed vs. Accident Severity



Detailed Data View

Conclusion

The traffic data analysis and visualization application successfully integrates advanced data processing techniques with interactive visualizations, providing a robust tool for traffic management and urban planning. By combining real-time data, predictive analytics, and interactive tools, it empowers users to make informed decisions based on comprehensive traffic and environmental insights. This application not only enhances the understanding of traffic dynamics but also supports proactive measures in traffic safety and urban development planning.