Janus

Navigate to Safety@NYC

Big Data

12/22/2023

Abhay Garg - ag9489

Pragnavi Ravuluri Sai Durga - pr2370

NYU

Contents

NYU

1. Problem Statement

Addressing Pedestrian Safety Concerns in NYC

New York City, renowned for its vibrant street life and dense urban landscape, presents a unique set of challenges for pedestrian safety. Recent polls have highlighted a concerning trend: a significant majority of pedestrians, approximately 70%, report feeling unsafe while navigating the city's streets. This pervasive sense of insecurity not only diminishes the quality of urban life but also poses a substantial public safety issue.

The genesis of this problem lies in a complex interplay of factors, including but not limited to, high traffic volumes, the behavior of motorists and pedestrians, urban design, and crime rates. The traditional approach to addressing pedestrian safety has predominantly focused on physical infrastructure improvements and law enforcement measures. However, these solutions, while beneficial, have not fully alleviated the underlying sense of insecurity experienced by pedestrians.

In response to this challenge, our project introduces an innovative solution: a web application designed to enhance pedestrian safety in New York City. This application aims to empower pedestrians with information that can significantly improve their sense of security and safety while navigating the city.

Our solution leverages the power of Big Data analytics and technology. By integrating the Google Maps API, we obtain optimal routing information for pedestrian travel. To augment this, we utilize the New York City Arrest data, processed through Spark, a powerful tool for big data analysis. This integration allows us to assign a risk score to each potential route based on historical crime data in the route.

In summary, this project seeks to bridge the gap between pedestrian safety needs and technological innovation, offering a novel approach to a persistent urban challenge. By providing pedestrians with safer route options, we aspire to enhance the overall pedestrian experience in New York City, making the streets not only feel safe but also more inviting for all.

2. Data Source

NYPD Arrest Data (Year to Date)

https://data.cityofnewyork.us/Public-Safety/NYPD-Arrest-Data-Year-to-Date-/uip8-fykc

Columns: [ARREST_KEY, ARREST_DATE, PD_CD, PD_DESC, KY_CD, OFNS_DESC, LAW_CODE, LAW_CAT_CD, ARREST_BORO, ARREST_PRECINCT, JURISDICTION_CODE, AGE_GROUP, PERP_SEX, PERP_RACE, X_COORD_CD, Y_COORD_CD, Latitude, Longitude]

Preview of NYPD Arrest Data (Year to Date)

ARREST_...	ARREST_...	PD_CD	PD_DESC	KY_CD	OFNS_DE...	LAW_CODE	LAW_CAT...	ARREST_...	ARREST_...	JURISDIC...	AGE_GRO...	PERP_SEX	PERP_RA...	X_COORD...	Y_COORD...	Latitude	Longitude	New Geor...
261182234	01/01/2023	101	ASSAULT 3	344	ASSAULT 3 & ...	PL 1200001	M	M	6	0	18-24	M	WHITE	983342	206250	40.732785	-74.003276	POINT (-74.00...
261182239	01/01/2023	339	LARCENY,PETI...	341	PETIT LARCE...	PL 1552500	M	M	6	0	45-64	M	WHITE	985152	204777	40.728745	-73.996745	POINT (-73.99...
261182241	01/01/2023	493	STOLEN PROP...	111	POSSESSION ...	PL 1654505	F	M	28	0	25-44	M	BLACK	997412	230102	40.79824299	-73.95246182	POINT (-73.95...
261185967	01/01/2023	105	STRANGULATI...	106	FELONY ASSA...	PL 1211200	F	Q	105	0	25-44	M	WHITE	1054755	203922	40.726116	-73.745626	POINT (-73.74...
261186604	01/01/2023	792	CRIMINAL PO...	118	DANGEROUS ...	PL 265031B	F	K	69	0	18-24	M	WHITE HISPA...	1015413	170673	40.635082	-73.887719	POINT (-73.88...
261186608	01/01/2023	101	ASSAULT 3	344	ASSAULT 3 & ...	PL 1200001	M	K	83	0	18-24	M	WHITE HISPA...	1008810	194859	40.70148558	-73.91142354	POINT (-73.91...
261186616	01/01/2023	397	ROBBERY,OPE...	105	ROBBERY	PL 1600500	F	M	28	0	25-44	M	BLACK	997412	230102	40.79824299	-73.95246182	POINT (-73.95...
261186617	01/01/2023	918	RECKLESS DR...	348	VEHICLE AND ...	VTL1212000	M	M	34	0	18-24	M	WHITE HISPA...	1004892	253548	40.86258121	-73.92537361	POINT (-73.92...
261186620	01/01/2023	705	FORGERY,ETC...	358	OFFENSES IN...	PL 1702000	M	Q	105	0	18-24	M	BLACK	1055041	186244	40.67759041	-73.74478108	POINT (-73.74...
261186622	01/01/2023	101	ASSAULT 3	344	ASSAULT 3 & ...	PL 1200001	M	Q	110	0	18-24	F	WHITE HISPA...	1022419	212521	40.749918	-73.862239	POINT (-73.86...
261186624	01/01/2023	109	ASSAULT 2,1,...	106	FELONY ASSA...	PL 1200502	F	Q	110	0	25-44	F	WHITE HISPA...	1020728	211337	40.746673	-73.868351	POINT (-73.86...
261187988	01/01/2023	101	ASSAULT 3	344	ASSAULT 3 & ...	PL 1200001	M	K	84	0	18-24	M	BLACK	988466	194673	40.701008	-73.984794	POINT (-73.98...
261188000	01/01/2023	268	CRIMINAL MI...	121	CRIMINAL MI...	PL 1450502	F	Q	110	0	25-44	M	BLACK HISPA...	1020728	211337	40.746673	-73.868351	POINT (-73.86...
261188320	01/01/2023	905	INTOXICATED ...	347	INTOXICATED ...	VTL119202U	F	K	76	0	25-44	M	UNKNOWN	985004	186837	40.67950109	-73.99728157	POINT (-73.99...
261188324	01/01/2023	101	ASSAULT 3	344	ASSAULT 3 & ...	PL 1200001	M	Q	105	0	25-44	M	WHITE HISPA...	1052086	200048	40.715504	-73.755295	POINT (-73.75...
261195432	01/01/2023	113	MENACING,U...	344	ASSAULT 3 & ...	PL 1201401	M	K	69	0	45-64	M	BLACK	1009538	171858	40.63835096	-73.90888428	POINT (-73.90...
261195433	01/01/2023	109	ASSAULT 2,1,...	106	FELONY ASSA...	PL 1200502	F	K	83	0	18-24	M	WHITE HISPA...	1009212	192148	40.694045	-73.909981	POINT (-73.90...

< Previous Next > Showing rows 1 to 17 out of 112,571

Rows - 170K
Columns - 19
Each row is a Arrest in NYC by NYPD

This dataset contains the current year's data up to the recent quarter (Jan 2023 to Sep 2023).

Out of 19 columns in the dataset, the significant columns required for computation in this project are ARREST_DATE, Latitude and Longitude. Rest of the columns are used to display metadata on the User Interface for an enhanced user experience.

This dataset is updated on a quarterly basis. However, this doesn't affect the output much considering minimal deviation in the risk score computer for the data over the past 17 years.

NYU

NYPD Arrest Data (Historic)

https://data.cityofnewyork.us/Public-Safety/NYPD-Arrests-Data-Historic-/8h9b-rp9u

Columns: [ARREST_KEY, ARREST_DATE, PD_CD, PD_DESC, KY_CD, OFNS_DESC, LAW_CODE, LAW_CAT_CD, ARREST_BORO, ARREST_PRECINCT, JURISDICTION_CODE, AGE_GROUP, PERP_SEX, PERP_RACE, X_COORD_CD, Y_COORD_CD, Latitude, Longitude, Lon_Lat]

ARREST_KEY	ARREST_DA…	PD_CD	PD_DESC	KY_CD	OFNS_DESC	LAW_CODE	LAW_CAT_CD	ARREST_B…	ARREST_PR…	JURISDICTI…	AGE_GROUP	PERP_SEX	PERP_RACE	X_COORD_CD	Y_COORD_CD	Latitude	Longitude	Lon_Lat
236791704	11/22/2021	581				PL 2225001	M	M			45-64	M	BLACK	997427	230378	40.799008797…	-73.952408540…	POINT (-73.95…
237354740	12/04/2021	153	RAPE 3	104	RAPE	PL 1302502	F	B	41	0	25-44	M	WHITE HISPAN…	1013232	236725	40.816391847…	-73.895296413…	POINT (-73.89…
236081433	11/09/2021	681	CHILD, ENDAN…	233	SEX CRIMES	PL 2601001	M	Q	113	0	25-44	M	BLACK	1046367	186986	40.679700408…	-73.776047367…	POINT (-73.77…
32311380	06/18/2007	511	CONTROLLED …	235	DANGEROUS D…	PL 2200300	M	Q	27	1	18-24	M	BLACK					
192799737	01/26/2019	177	SEXUAL ABUSE	116	SEX CRIMES	PL 1306503	F	M	25	0	45-64	M	BLACK	1000555	230994	40.800694331…	-73.941109285…	POINT (-73.94…
193260691	02/06/2019					PL 2203400	F	M	14	0	25-44	M	UNKNOWN	986685	215375	40.757839003…	-73.991212110…	POINT (-73.99…
237291769	12/03/2021	579				PL 2224001	F	Q	115	0	25-44	M	BLACK	1018534	220579	40.772056496…	-73.876224000…	POINT (-73.87…
236106641	11/10/2021	263	ARSON 2,3,4	114	ARSON	PL 1501001	F	B	41	72	25-44	M	WHITE HISPAN…	1017934	232221	40.804012949…	-73.878331832…	POINT (-73.87…
238383628	12/28/2021	729	FORGERY,ETC.,…	113	FORGERY	PL 1702500	F	Q	113	0	18-24	M	BLACK	1045482	191341	40.691660017…	-73.779198520…	POINT (-73.77…
149117452	01/06/2016	153	RAPE 3	104	RAPE	PL 1302503	F	K	67	0	25-44	M	BLACK	998032.0	175598.0	40.648650085…	-73.950335562…	POINT (-73.95…
237339209	12/04/2021	101	ASSAULT 3	344	ASSAULT 3 & R…	PL 1200001	M	K	83	0	25-44	M	BLACK	1007400	190154	40.688583516…	-73.916526346…	POINT (-73.91…
221756278	12/12/2020					PL 2203400	F	M	23	0	25-44	M	WHITE HISPAN…	999998	226211	40.787567301…	-73.943132331…	POINT (-73.94…
237580757	12/09/2021	105	STRANGULATI…	106	FELONY ASSA…	PL 1211200	F	M	30	0	25-44	M	BLACK	999751	241188	40.828675458…	-73.943989715…	POINT (-73.94…
190049060	11/15/2018	157	RAPE 1	104	RAPE	PL 1303501	F	K	77	0	25-44	M	BLACK	1003606	185050	40.674583308…	-73.930221540…	POINT (-73.93…
24288194	09/13/2006	203	TRESPASS 3, C…	352	CRIMINAL TRE…	PL 140100E	M	K	77	2	45-64	M	BLACK	1004580.0	183838.0	40.671254457…	-73.926713851	POINT (-73.92…
221870158	12/15/2020	297	FACILITATION …	354	ANTICIPATORY…	PL 1150000	M	K	75	0	45-64	F	BLACK	1019745	184405	40.672762932…	-73.872042636…	POINT (-73.87…
220422940	11/12/2020	157	RAPE 1	104	RAPE	PL 1303502	F	Q	112	0	25-44	M	BLACK	1025420	202485	40.722363687…	-73.851473893…	POINT (-73.85…
237954587	12/16/2021	397	ROBBERY,OPE…	105	ROBBERY	PL 1600500	F	M	1	0	25-44	M	WHITE	982351	201758	40.720463840…	-74.006852203…	POINT (-74.00…
189182271	10/24/2018	153	RAPE 3	104	RAPE	PL 1302503	F	M	5	0	45-64	M	WHITE HISPAN…	984946	200203	40.716195914…	-73.997490745…	POINT (-73.99…
222293770	12/27/2020	681	CHILD, ENDAN…	233	SEX CRIMES	PL 2601001	M	B	43	0	25-44	M	BLACK	1020316	239179	40.823101299…	-73.869690460…	POINT (-73.86…
214798896	07/01/2020	177	SEXUAL ABUSE	116	SEX CRIMES	PL 1306504	F	B	46	0	25-44	M	BLACK	1011881	250411	40.853960274…	-73.900120874…	POINT (-73.90…
237622840	12/10/2021	397	ROBBERY,OPE…	105	ROBBERY	PL 1601503	F	M	6	0	25-44	M	BLACK	982746	206647	40.733883033…	-74.005428310…	POINT (-74.00…
196324211	04/23/2019	157	RAPE 1	104	RAPE	PL 1303501	F	K	77	0	45-64	M	BLACK HISPA…	1003606	185050	40.674583308…	-73.930221540…	POINT (-73.93…
196785901	05/04/2019	175	SEXUAL ABUS…	233	SEX CRIMES	PL 13052A1	M	B	50	0	25-44	M	BLACK	1011257	261130	40.883382579…	-73.902333308…	POINT (-73.90…
236176798	11/12/2021	681	CHILD, ENDAN…	233	SEX CRIMES	PL 2601001	M	M	25	0	25-44	M	BLACK	1000555	230994	40.800694331…	-73.941109285…	POINT (-73.94…
197554056	05/23/2019	175	SEXUAL ABUS…	233	SEX CRIMES	PL 13052A1	M	M	26	0	45-64	F	BLACK	996241	236149	40.814850289…	-73.956681847…	POINT (-73.95…
189129210	10/23/2018	101	ASSAULT 3	344	ASSAULT 3 & R…	PL 1200000	M	M	18	0	25-44	M	WHITE HISPAN…	987501	217778	40.764434359…	-73.988265529	POINT (-73.98…
237856894	12/15/2021	101	ASSAULT 3	344	ASSAULT 3 & R…	PL 1200001	M	B	43	0	25-44	M	BLACK	1021043	241958	40.830725856…	-73.867048483…	POINT (-73.86…
220303765	11/10/2020	153	RAPE 3	104	RAPE	PL 1302501	F	Q	112	0	25-44	M	BLACK	1025420	202485	40.722363687…	-73.851473893…	POINT (-73.85…
238359804	12/28/2021	681	CHILD, ENDAN…	233	SEX CRIMES	PL 2601001	M	K	84	0	45-64	M	BLACK	989013	192652	40.695468941…	-73.982825078…	POINT (-73.98…
189827117	11/10/2018	175	SEXUAL ABUS…	233	SEX CRIMES	PL 13052A1	M	M	1	0	25-44	M	WHITE HISPAN…	982285	201682	40.720255223…	-74.007090279…	POINT (-74.00…
189176315	10/24/2018	475				PL 1651601	M	M	28	1	25-44	M	BLACK	999346	233573	40.807775126…	-73.945470282…	POINT (-73.94…
238116432	12/21/2021	579				PL 2224002	F	Q	113	0	25-44	M	WHITE HISPAN…	1046405	187113	40.680048726…	-73.775909193…	POINT (-73.77…
237302061	12/03/2021					PL 2203400	F	M	14	0	25-44	M	BLACK HISPA…	986195	213562	40.752862899…	-73.992981331…	POINT (-73.99…
238376572	12/28/2021	579				PL 2223501	F	K	72	0	25-44	M	ASIAN / PACIF…	981639	172112	40.639092101…	-74.009409294…	POINT (-74.00…
220917769	11/24/2020	157	RAPE 1	104	RAPE	PL 1303501	F	B	41	0	25-44	M	BLACK	1013232	236725	40.816391847…	-73.895296413…	POINT (-73.89…
236918575	11/24/2021	153	RAPE 3	104	RAPE	PL 1302502	F	Q	114	0	25-44	M	WHITE	1007654	219564	40.769306087…	-73.915508179…	POINT (-73.91…
238350561	12/28/2021	157	RAPE 1	104	RAPE	PL 1303501	F	B	41	0	25-44	M	WHITE HISPAN…	1013232	236725	40.816391847…	-73.895296413…	POINT (-73.89…

< Previous Next >

Showing Arrest in NYC by NYPDs 1 to 100 out of 5,498,65

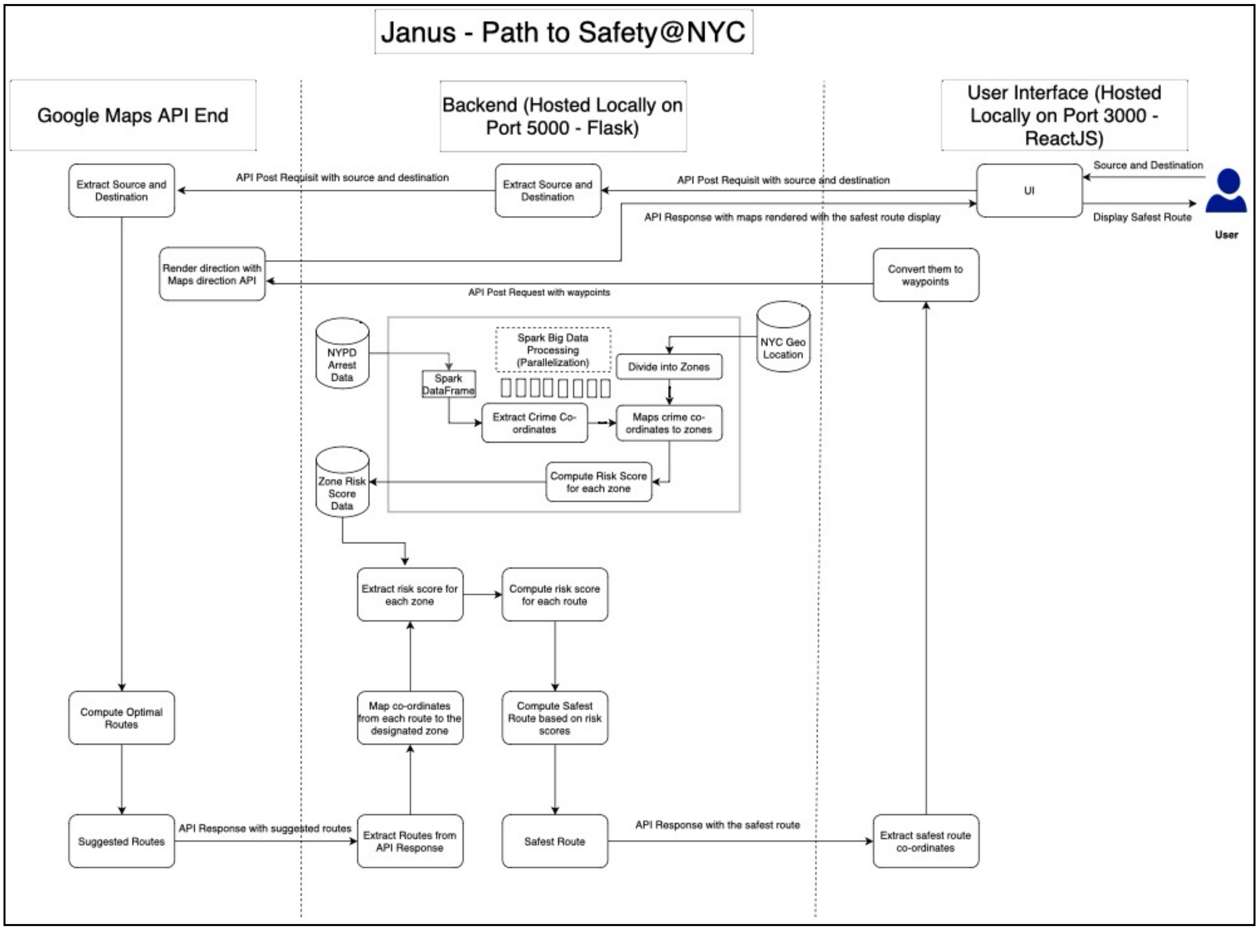
Rows - 5.5M
Columns - 19
Each row is a Arrest in NYC by NYPD

This dataset contains the past years' data starting from the year 2006 up to the previous year (Jan 2006 to Dec 2022).

Similar to the first one, out of 19 columns in the dataset, the significant columns required for computation in this project are ARREST_DATE, Latitude and Longitude. Rest of the columns are used to display metadata on the User Interface for an enhanced user experience.

This dataset is updated on an annual basis. Considering the amount of parallelization required to process the data and perform operations on the dataset, this data is perfect for a Big Data application project.

3. Architecture (High Level)



Description

The architecture diagram outlines the high level project design and highlights the technologies used to achieve an optimal solution to the problem statement. The system is structured into three main components: the locally hosted User Interface developed using ReactJS, the locally hosted Backend with Flask and Spark and the Google Maps API.

Let's now discuss each component on high level:

Google Maps API: This segment is responsible for interacting with the Google Maps API to extract the optimal routes between given source and destination. It sends an API response to the backend with the suggested routes for a pedestrian. Subsequently, the direction API renders the suggested safest route into the Google Maps interface upon receiving a request from the user interface.

Backend (Spark, Flask): The Backend serves as the core processing unit of the system. It receives the suggested routes from the Google Maps API and employs Spark for big data processing, utilizing the NYPD Arrest Data to pick the safest route among them. Spark DataFrame operations are performed to extract crime coordinates, which are then mapped into zones with their respective computed risk scores. The Backend computer risk scores for each zone and each route, ultimately determining the safest route based on the accumulated risk scores.

User Interface (ReactJS): The user interface is the front-end component where users input their source and destination. The UI then sends a request to the Backend API with the source and destination values and receives the safest route for a response. The safest route coordinates are then attached to the Google Maps Directional API request as waypoints and from the received response UI displays the safest route returned by the Backend. UI is hosted locally and it provides a user-friendly interface for the system's interaction with end-users.

Overall, the architecture facilitates a seamless flow of data between the user's input and the delivery of the safest possible route. The application strategically combines the functionalities of Google Maps and Spark to deliver a robust solution aimed at improving pedestrian safety in NYC.

2. Methodology

Safest Route Calculation

1. Define the grid of NYC based on minimum and maximum latitude and longitude of NYC. The grid consists of 5000 rows and 5000 columns leading to 25 million zones.

2. Calculate the step-size for latitude and longitude based on the specified number of divisions.

num_divisions = 5000
lat_step = (max_latitude - min_latitude) / num_divisions
lon_step = (max_longitude - min_longitude) / num_divisions

3. Create a function named find_zone_id which takes two parameters latitude and longitude and returns the zone_id.

lat_index = int((latitude - min_latitude) / lat_step)
lon_index = int((longitude - min_longitude) / lon_step)
zone_id = lat_index * num_divisions + lon_index

```
# Define the new boundaries according to data
min_latitude, max_latitude = 40.49, 62.08
min_longitude, max_longitude = -74.26, -73.68

# Number of divisions along each axis to create 25,000,000 zones (5000x5000)
num_divisions = 5000

lat_step = (max_latitude - min_latitude) / num_divisions
lon_step = (max_longitude - min_longitude) / num_divisions
```

```
from pyspark.sql.functions import udf
from pyspark.sql.types import IntegerType

def find_zone_id(latitude, longitude):
    # Input validation
    if not (min_latitude <= latitude <= max_latitude) or not (min_longitude <= longitude <= max_longitude):
        return "Invalid latitude or longitude"

    # Calculate indexes
    lat_index = int((latitude - min_latitude) / lat_step)
    lon_index = int((longitude - min_longitude) / lon_step)

    # Handle edge cases
    if lat_index == num_divisions:
        lat_index -= 1
    if lon_index == num_divisions:
        lon_index -= 1

    # Calculate zone_id
    zone_id = lat_index * num_divisions + lon_index
    return zone_id

find_zone_id_udf = udf(find_zone_id, IntegerType())
```

4. Now, in the arrest dataframe, call the UDF find_zone_id to calculate the zone_id for each record. The data frame will look like this:

```
+-----------+-----------+---------+--------+--------------+---------+----------+-------+
|ARREST_DATE|ARREST_BORO|AGE_GROUP|PERP_SEX|     PERP_RACE| Latitude| Longitude|zone_id|
+-----------+-----------+---------+--------+--------------+---------+----------+-------+
| 2021-11-22|          M|    45-64|       M|         BLACK|40.799009|-73.952409| 357651|
| 2021-12-04|          B|    25-44|       M|WHITE HISPANIC|40.816392|-73.895296| 378144|
| 2021-11-09|          Q|    25-44|       M|         BLACK|  40.6797|-73.776047| 219172|
| 2019-01-26|          M|    45-64|       M|         BLACK|40.800694|-73.941109| 357749|
| 2019-02-06|          M|    25-44|       M|       UNKNOWN|40.757839|-73.991212| 312317|
| 2021-12-03|          Q|    25-44|       M|         BLACK|40.772056|-73.876224| 328308|
| 2021-11-10|          B|    25-44|       M|WHITE HISPANIC|40.804013|-73.878332| 363290|
| 2021-12-28|          Q|    18-24|       M|         BLACK| 40.69166|-73.779199| 234144|
| 2016-01-06|          K|    25-44|       M|         BLACK| 40.64865|-73.950336| 182669|
| 2021-12-04|          K|    25-44|       M|         BLACK|40.688584|-73.916526| 227960|
+-----------+-----------+---------+--------+--------------+---------+----------+-------+
only showing top 10 rows
```

5. To calculate the risk score of each zone, calculate the total number of crimes per zone on each arrest date. Then, Risk-Score for a zone will be the average number of crimes per day in that zone.

```python
# Calculate the total number of crimes per zone on each arrest date
crime_count_per_day_zone = new_df_with_zone_id.groupBy("zone_id", "ARREST_DATE").agg(count("*").alias("daily_crimes"))

# Calculate the average crime rate per day for each zone
crime_rate_per_zone = crime_count_per_day_zone.groupBy("zone_id").agg(avg("daily_crimes").alias("risk_score"))
crime_rate_per_zone.dropna()
```

zone_id	risk_score
213516	1.0
279131	1.0
448531	1.2837837837837838
417823	1.2075471698113207
197258	1.2133620689655173
423444	1.1428571428571428
393144	1.2833333333333334
438279	1.4070996978851964
313148	1.625

6. Save the dataframe into a csv file and convert it to a dictionary where key is zone id and value is risk score. This will help to fetch the risk score of a zone in O(1) time.

8. Now, given a source address and a destination address, google map routes API returns the optimal routes with the coordinates. Then, calculate the risk score for each route and return the route with least risk-score which is the safest route to follow. To calculate the risk score for a route, map its coordinates with zone-ids. The final risk score for this route is the summation of all the risk scores of all its zones.

APIs

1. GET Routes Risk Score API

Endpoint: GET /get-routes-risk-score

Query Parameters:

source (string): Starting point of the route.

destination (string): Ending point of the route.

Response:


```
{
  "status": "SUCCESS",
  "message": "GetRoutesRiskScore Api Handler",
  "routes": {
    "0": {
      "risk_score": 1.0797387773682987,
      "distance": "0.7 mi",
      "time": "15 mins",
      "Coordinate": [
        {"lat": 40.7296912, "long": -73.997006},
        {"lat": 40.7295523, "long": -73.996668},
        // ... (additional coordinates)
      ]
    },
    "1": {
      "risk_score": 1.2778759937747266,
      "distance": "0.7 mi",
      "time": "17 mins",
      "Coordinate": [
        {"lat": 40.7296912, "long": -73.997006},
        {"lat": 40.7295623, "long": -73.9967441},
        // ... (additional coordinates)
      ]
    },
    // ... (additional routes)
  }
}
```


Description: The API calculates and returns multiple routes between the specified source and destination along with their associated risk scores, distances, and estimated times. Each route contains a risk score, distance, time, and a list of coordinates comprising latitude and longitude.

2. GET Safest Route API

Endpoint: GET /get-safest-route

Query Parameters:

source (string): Starting point of the route.
destination (string): Ending point of the route.

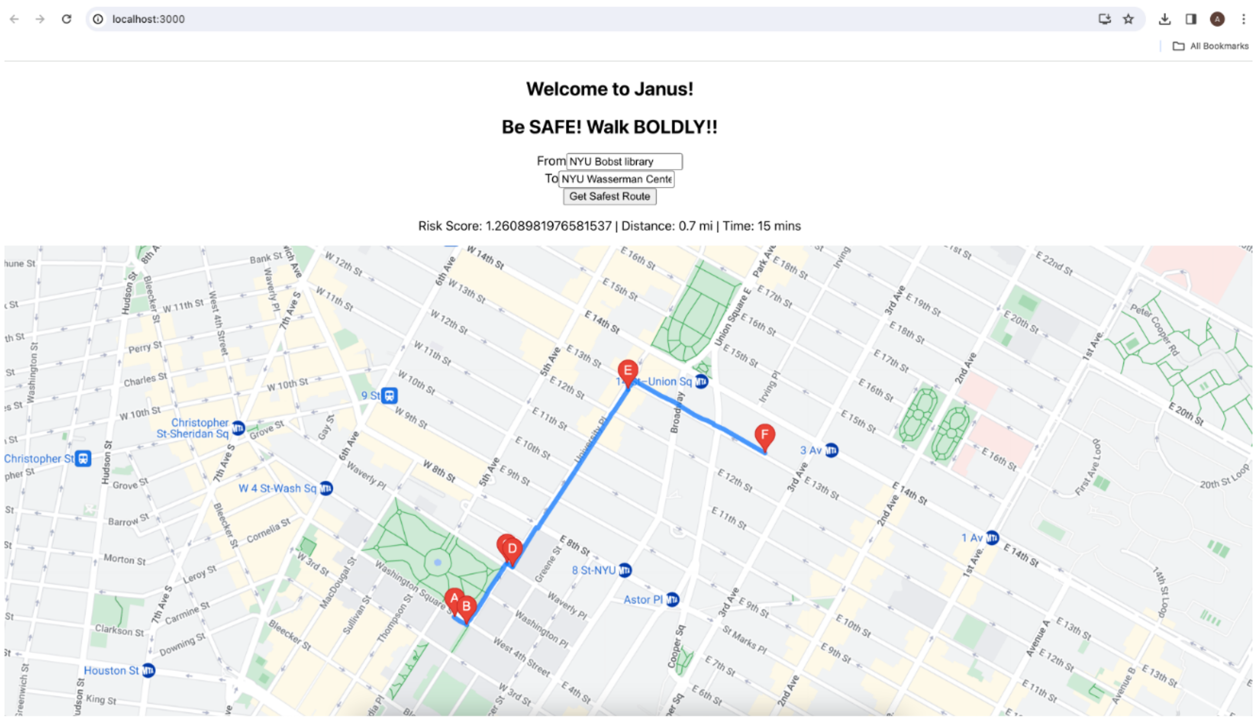
Response:


```
{
  "status": "SUCCESS",
  "message": "GetRoutesRiskScore Api Handler",
  "route": {
    "risk_score": 1.0797387773682987,
    "distance": "0.7 mi",
    "time": "15 mins",
    "Coordinate": [
      {"lat": 40.7296912, "long": -73.997006},
      {"lat": 40.7295523, "long": -73.996668},
      // ... (additional coordinates)
    ]
  }
}
```


Description:

This API calculates and returns the safest route between the specified source and destination, providing information such as risk score, distance, time, and the list of coordinates. The response includes details for the single safest route based on the risk score.

User Interface

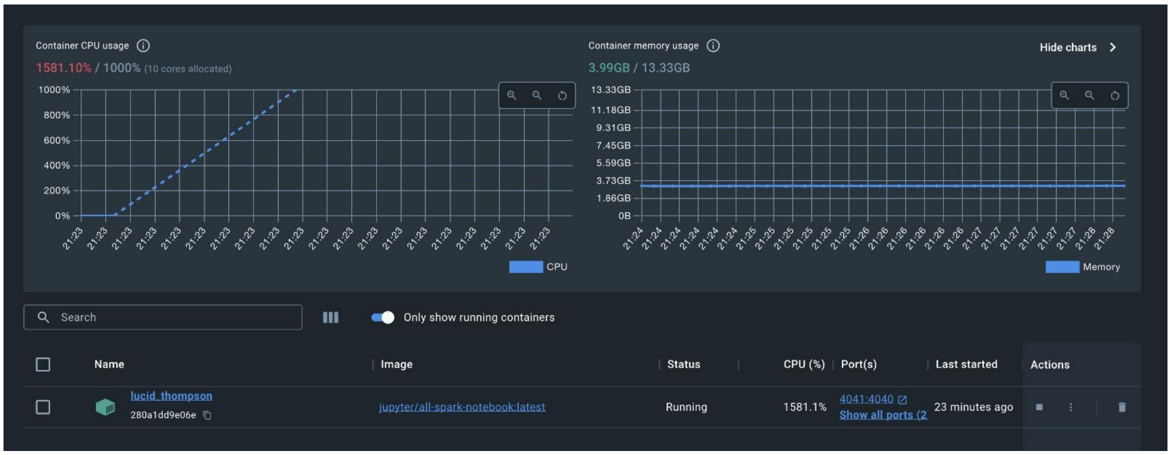


Upon receiving the response from API endpoint, it shows the route on google map using Google Maps' Direction Service React API where the route coordinates (waypoints) are passed to draw a map.

5. Optimization

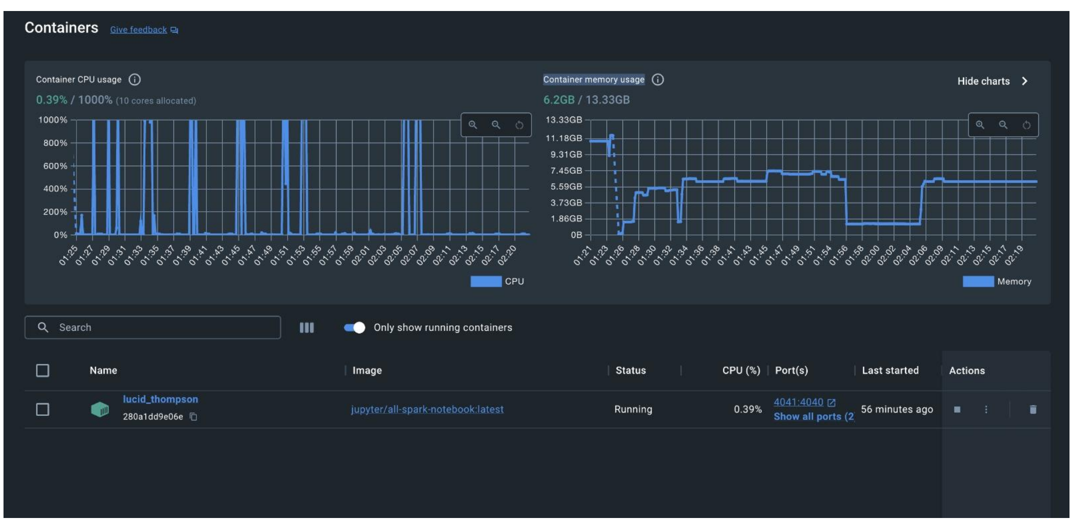
Challenge

The initial approach to map coordinates for over 5 million records to 250,000 zones using a join operation has proven to be inefficient and suboptimal. It has been running for 4 hours, and there is no end in sight.



Workaround

To address the inefficiency, an alternative and highly efficient approach was adopted. It involved mapping coordinates for the same 5 million records, but this time to 25 million zones using a User-Defined Function (UDF). Despite having the same available resources of 14 GB memory and 10 cores, this workaround took only approximately 1 minute and 20 seconds to complete the task.



6. Future Scope

1. A better user interface which also shows the alternative routes to take with their risk score.
2. Update NYC arrest data timely using Spark Streaming technologies.
3. Integrating with existing maps (Google Maps, OpenstreetMap, etc.)

7. Source Code

Github Link: https://github.com/gargabhay1999/Janus-BigData