Traffic Collision Analytics: A Data-Driven Approach to Road Safety

Dilochan Karki, Yunik Tamrakar

April 28, 2024

1 Introduction

Approximately 1.19 million people die each year as a result of traffic related accidents. It is the leading cause of death among people aged 5 to 29 [9]. Thus, reducing traffic accidents is an important public safety challenge. Accident prediction and analysis is important for optimizing public transportation, enabling safer routes, and improving the road infrastructure, all with the ultimate goal of making the roads safer. Performing holistic analytics on traffic accident datasets is one of the most effective and efficient methods to help reduce damages due to such accidents. For the analytics to be as effective as possible, it is necessary for the dataset to be very comprehensive. Our work makes use of the largest public dataset on traffic accidents : US-Accidents [5]. US-Accidents currently contains data from more than 7 million instances of traffic accidents that took place within the contiguous United States, from 2016 - 2023. We perform two types of analytics on this national dataset - the first being aggregate analysis on all data from 2016 to 2023 and the second being yearly analysis of the accidents ranging from 2016 to 2023. Besides, in order to perform our analysis on a more granular level, we also perform analysis on two more localized datasets. One of these datasets contains collision data from New York city and the other contains data from Chicago. Our analysis involves several components such as analyzing when most accidents happen, where most accidents happen and also various other conditions that can influence accidents such as weather conditions, wind properties and so on. Our work also includes a visualization component for our findings so as to provide the stakeholders and the readers with a more intuitive and concise presentation of our findings. We also make use of the rich accident coordinates information present in the datasets and thus, plot many of our findings on real maps. Last, but not the least, we also make use of distributed computation and distributed storage frameworks in our work for efficient analysis of the datasets.

We believe our work will be of great interest to the engineers, researchers and policy makers in the urban planning and transportation engineering space as well as the geo-spatial analysis space. Besides, our work should also be of technical interest to people working in the data science and distributed computation domain.

2 Problem Characterization

One of the key challenges with traffic collision analytics is the collection of accurate and comprehensive accident related data. Several details need to be accounted for in the dataset such as the exact location, time, involved vehicles, weather conditions, road conditions, and behavior of drivers and pedestrians. Also, different sources, such as police reports, insurance claims, and witness accounts, often provide conflicting or incomplete data. As is the case with most form of analytics, the quality of the analysis is heavily dependent on the quality of the data that is being analyzed. Thus, it is imperative that the data that is collected be of high quality. Luckily for us, due to excellent work of the researchers before us, we were able to leverage these carefully curated datasets made available to the public.

However, these datasets are not perfect. Several key data points are missing from the datasets (particularly from the US dataset). One of the glaring omissions we noted was the absence of accidents data from NYC. Also, there can be several social challenges associated with this problem space as well. The collected data often includes personally identifiable information (PIIs), which must be handled within the strict confines of legal frameworks like GDPR. Furthermore, public acceptance

of surveillance technologies and data collection methods is also crucial, as community resistance can impede the implementation of such analytics systems.

After issues related to data collection, data cleanup, we are faced with the key question of what data points to analyze. After a careful study of the problem space, we narrowed down our primary questions to - when and where do accidents happen. After deriving insights pertaining to these exploratory questions, we were also able to ask questions as to why accidents happen for a given set of conditions. Ultimately, we attempt to answer these explanatory questions based on data patterns as well as experience. The datasets we analyze for our work also come with computational and algorithmic challenges. Single node analytics as well as storage of these heavy datasets can prove to be impossible due to resource constraints and may require several alternate strategies such as data chunking. Also, the intermediate data structures generated during processing such large datasets can be very heavy and thus, requires alternate caching strategies which will be discussed in the subsequent sections.

3 Dominant Approaches to the Problem

There have been other works on analysis of traffic accidents but those have been performed on smaller datasets and don't provide a comprehensive outlook on the attributes and sources of traffic accidents. [6] A holistic approach to looking at traffic accidents analysis is required that can provide information on causes, consequences and environments of the accidents. The datasets we used for our experiments have been used by previous researchers [5] to perform thorough analysis through extensive experiments using a deep neural network to predict rare accident events. Despite having sparse and hard to obtain data, they were able to consolidate data based on various attributes such as traffic events, weather and time data. This allowed them to analyze different categories for traffic accident and found that points-of-interest and traffic events had a significant impact. However, they don't incorporate data from wide range of sources to further study other possible features that could have correlations with traffic accidents. Similarly, they don't explain about the storage and computation cost to process such a wide collection of heterogeneous data. In addition to that, the current size of the data since their experiments has doubled which demands further exploration into means of storing and processing such a large data for analytics and predictive learning. There also are other works like [4] that provide information on various data sources, algorithms and techniques to provide accident forecasts.

Scholars like us have mostly used libraries and solutions based on single-node computing for data analytics. Common solutions include using Pandas and Matplotlib libraries in Python to perform exploratory data analysis and visualization. However, this dataset provided us with a challenge to take a step beyond and use existing distributed computing approaches for efficient handling of large heterogeneous datasets.

4 Methodology

Our plan of attack implemented in the project can be broadly divided into the following steps:

- Data Collection
- Data Exploration
- Data Storage
- Distributed Data Analytics
- Data Visualization

4.1 Data Collection

As mentioned earlier, we use the US-Accidents datasets for our aggregate and yearly analysis of the crashes across the US. The original paper based on the dataset contained accident records up to 2019. However, the dataset was maintained and augmented and data until 2023 was made available under open source license at Kaggle. We use this comprehensive version of the dataset, which contains over 7 million records spanning from 2016-2023. For our analysis of the NYC and Chicago crashes, we use

the Motor Vehicle Crashes - NYC and Motor Vehicle Crashes - Chicago dataset respectively, both made available on data.gov.

4.2 Data Exploration

We perform some basic data exploration using the Pandas framework. We do some basic operations on the dataset such as counting the total number of records, missing records and so on. A key observation made during the EDA phase was that the US-Dataset did not contain accident records from NYC, which led us into searching for a separate dataset for NYC crashes. We used jupyter notebook as well as Google Collaboratory to perform the exploratory data analytics.

4.3 Data Storage

Our datasets are massive and thus, required us to perform additional measures to ensure proper storage and distribution of our dataset. We first ran into issues due to our large file size while pushing our data to Github. Github blocks file sizes that are greater than 100 MB. So, we had to use Git Large File Storage (LFS) [3] to push this dataset to our git repository. Also, for use with our spark cluster, we use the distributed capabilities of HDFS. HDFS can handle extremely large data sets by distributing the data across multiple nodes in a cluster, all in a fault tolerant manner.

4.4 Distributed Data Analytics

We use the Apache Spark framework to perform distributed data analytics. Our datasets are stored in a distributed manner using HDFS. We distribute our computations across multiple nodes in the CSB cluster. Our datasets consist of almost 10 million rows in total and occupy several gigabytes of disk space. Thus, it was imperative that we leveraged the computation and storage capabilities of multiple nodes to perform our analytics.

Our spark program is written in Scala. We use Scala over a language such as python to make use of the JVM related optimizations and to avoid dealing with the IPC overheads associated with using something like PySpark. We also use SparkSQL to generate our insights. The main reason why we choose SparkSQL over the native RDDs is the ease of use SparkSQL and the ability for us to write our SQL analytics queries without having to worry about the subtleties of converting it into the underlying RDDs. For our analytics, we ask two primary questions: when accidents happen and where accidents happen.

4.4.1 Spark related optimizations

As mentioned numerous times in this report, our datasets are fairly massive and thus, the dataframes generated from our datasets tend to be of large sizes as well. Not only that, we noticed that the computation of these dataframes took a significant amount of time. Thus, to mitigate the overhead of creating the dataframes for each lifecycle of the spark application, we decided to cache the dataframes on disk as Apache Parquet files. Parquet is a free, open-source file format that stores data in columns instead of rows. It's designed to be efficient for storing and retrieving large amounts of data and thus, seemed perfect for our use case. We use the gzip compression codec for the parquet files. We chose gzip because it offers a good balance between compression ratio and speed. For the US dataset, we have 8 dataframes corresponding to the 8 years between 2016 and 2023 and we create an Apache parquet cache for each of these years. This is a one time operation only and the subsequent lifecycles of the spark application can automatically use load the dataframes from the parquet format and thus, avoid the overhead of recomputing the dataframes. Furthermore, during the lifecycle of the application, we also use the concept of caching the dataframes in memory to reuse the dataframes as much as possible across computations.

When accidents happen?

We investigate

- the day of the week, month and the years when most accidents happen.

- the weather, wind, temperature conditions when most accidents happen

Where accidents happen?

We investigate

- states where most accidents happen
- cities where most accidents happen
- Interstate accidents heatmap
- accident hotspots for Colorado
- accident hotspots for Denver, CO
- accident hotspots for Fort Collins, CO

4.5 Data Visualization

We use Folium, Matplotlib and Seaborn to perform our visualization tasks. We codify the dataframes we obtain from our analytics into python code and then plot them into the graphs. For plotting coordinate related information such as accident hotspots, we use folium. Whereas, for plotting the statistics pertaining to accident counts, we use matplotlib and Seaborn. We also experimented with using KeplerGL to create heatmaps for the US accidents dataset. We noticed that folium would fail after the number of datapoints to be plotted onto the map exceeded 20k. However, KeplerGL did not run into such limitations, which establishes it as a robust solution for visualizing a very high number of datapoints.

5 Experimental Benchmarks

We benchmarked the amount of time it took to complete the entire Apache Spark execution and complete the SparkSQL query execution of USA, NYC and Chicago traffic datasets. The execution duration for these different datasets reveals notable differences in query execution times. For the NYC dataset, it can be noticed from 2 that the average query execution duration is at 0.15 seconds for each of the 6 queries. In contrast, the Chicago dataset exhibits a slightly faster query execution time for some queries giving an average of 0.11 seconds for 8 queries3. These findings suggest that the Chicago dataset queries were more quickly processed by SparkSQL compared to the NYC dataset, indicating impact of data size on performance since size of Chicago dataset was less than NYC dataset. Similarly, execution duration on the US dataset showcase considerably longer query execution times for rankings based on states compared to other queries1. This might be because of the aggregation of larger number of rows and attributes for each state. This substantial increase in processing time suggests impact of extensive dataset and more complex queries on processing time, highlighting potential challenges in handling larger-scale data with SparkSQL.

Furthermore, we validated the SparkSQL output by comparing the results with outputs produced with Pandas. For this, we wrote Python programs using Pandas library to read the large csv files in chunks and produce dataframes. We compared the outputs of programs for similar queries with results from SparkSQL. Likeness in these outputs validated our SparkSQL approach. However, running these programs wasn't possible in our personal computer due to large size of the csv file, despite reading the files as chunks with Pandas. Hence, for this purpose we ran those experiments in Google Colab notebooks.

5.1 Pros of Distributed Setup

As mentioned above, we tried running experiments in Pandas on our personal computers to validate the SparkSQL query outputs. Due to the size of the csv files and the limited memory in our laptops, the programs froze and couldn't produce an output. We tried running the programs in Jupyter Notebooks through CSB Lab computers as well but had similar issues, even while reading the csv in chunks.

Table 1: USA Dataset Query Execution Times

Query Name	Time Taken (s)
Severity of accidents (2016 - 2023)	0.022352
States ranked by Accident Severity Ratings	0.060132
Ranking by hour when most accidents happen	0.019690
Ranking by day of the week when most accidents happen	0.016173
Ranking by cities where most accidents happen	0.046788
Ranking by states where most accidents happen	0.016028
Ranking by years when most accidents happened	0.008608
Top 20 Temperature Ranges During Accidents	0.013166
Top 20 Weather Conditions During Accidents	0.010625
Top 10 States for Accidents During Heavy Rain	0.008731
Top 10 States for Accidents During Snow	0.008790
Top 10 States for Accidents During Mostly cloudy conditions	0.008389
Top 10 States for Accidents During Fair conditions	0.009030
Top 10 Weather Conditions for Accidents in Colorado	0.008842
Top 10 Wind Directions for Accidents in Colorado	0.009437
Top 10 Weather Conditions for Accidents on I-70 in Colorado	0.007899
Top 10 Interstates for Accidents	0.017343
Top 5 States for Accidents on I-95	0.007729
Top 5 States for Accidents on I-10	0.007248
Top 5 States for Accidents on I-5	0.007409
Top 5 States for Accidents on I-75	0.007166
Top 5 States for Accidents on I-80	0.007148
Top 10 Interstates for Snow Condition Accidents	0.011557
Top 5 States for Snow or Blizzard Conditions Accidents on I-90	0.006973
Top 5 States for Snow or Blizzard Conditions Accidents on I-94	0.006949
Top 5 States for Snow or Blizzard Conditions Accidents on I-80	0.006784
Top 5 States for Snow or Blizzard Conditions Accidents on I-35	0.006707
Top 5 States for Snow or Blizzard Conditions Accidents on I-70	0.006409

Table 2: NYC Dataset Query Execution Times

Query Name	Time Taken (s)
Number of Collisions Per Year	0.147123
Number of Crashes Per Day of the Week	0.127013
Number of Crashes Per Hour of the Day (Ranked)	0.142196
Number of Collisions Involving Pedestrians Per Year	0.120808
Number of Collisions by Borough and Year	0.139123
Number of Crashes with Different Vehicle Codes	0.119205

Table 3: Chicago Dataset Query Execution Times

Query Name	Time Taken (s)
Number of Collisions Per Year	0.143460
Number of Crashes Per Day of the Week	0.135545
Number of Crashes Per Hour of the Day (Ranked)	0.120803
Number of Crashes Per Weather Condition	0.135208
Number of Crashes Per Crash Type	0.062102
Number of Crashes Per Road Defect	0.029130
Number of Crashes per Injury Type	0.145830
Number of Crashes per Primary Cause	0.131011

We didn't experience any such issues while using Apache Spark for these datasets. We were able to successfully run SparkSQL programs even locally without any issues. This allowed to us to experience benefits of using a distributed computation framework to process large datasets efficiently and produce desired results in a quick manner with less hassle.

6 Insights Gleaned

Our comprehensive analytics revealed several interesting insights about the nature of accidents on a national level as well as on a regional level across heavily populated cities like NYC and Chicago.

6.1 When do accidents happen?

Our aggregate analysis of the US accidents dataset (from 2016-23) revealed that most accidents during the weekdays. We noticed that the frequency of accidents gradually increased from Monday and maxed out on Fridays. The weekends had the least number of accidents, which is to be expected given that many stay at home during then. Also, our analysis shows that the number of accidents during the summer time was much smaller than the number during the winter season. Figure 2 illustrates that the number of accidents start increasing as the fall season begins, peaks during December and again hits a downward trend as spring returns and things warm up. Also, figure 3 shows that 2021 was the year when most accidents happened, followed by 2022 and 2020. The graph shows an absurdly small number of crashes for 2023, which indicates that the data collection for the year may have been incomplete.

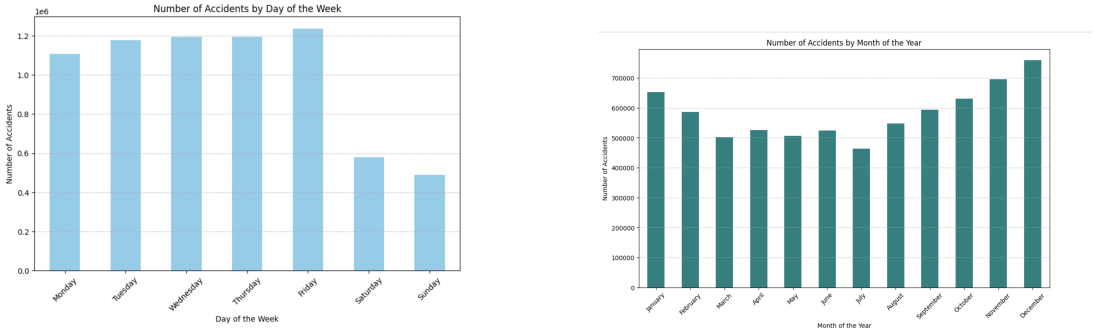


Figure 1: Left: Days when most accidents happen (2016-23). Right: Months when most accidents happen (2016-23).

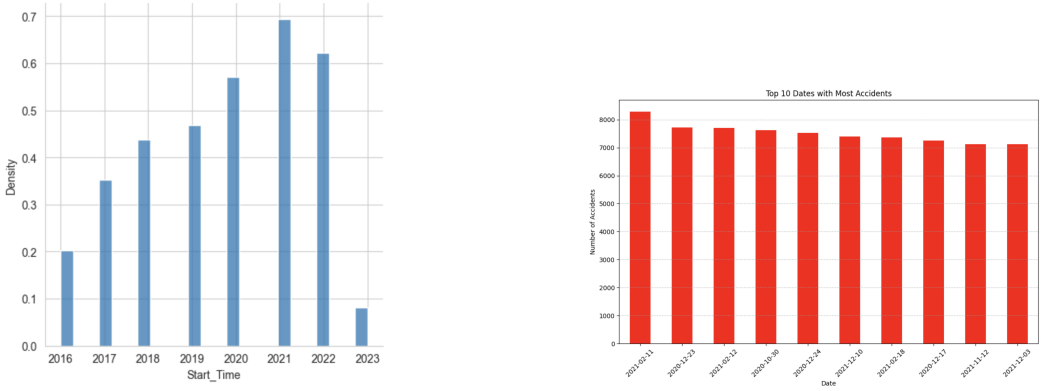
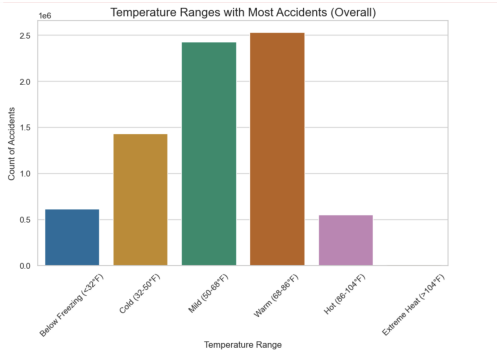


Figure 2: Left: Years when most accidents happened (2016-23). Right: Top dates when most accidents happened (2016-23).



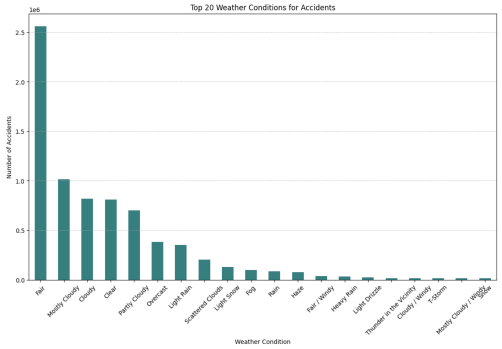
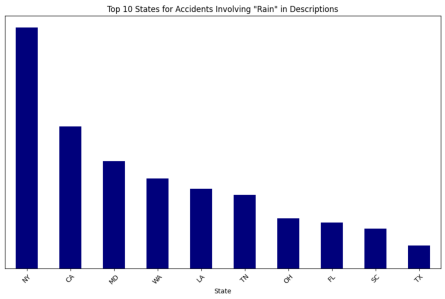


Figure 3: Left: Temperatures when most accidents happen (2016-23). Right: Weather conditions when most accidents happen (2016-23).



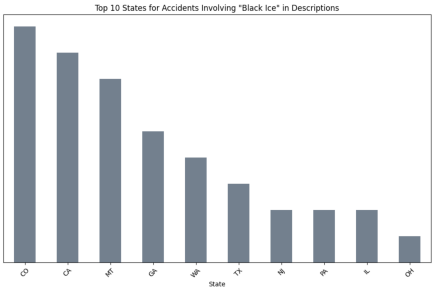
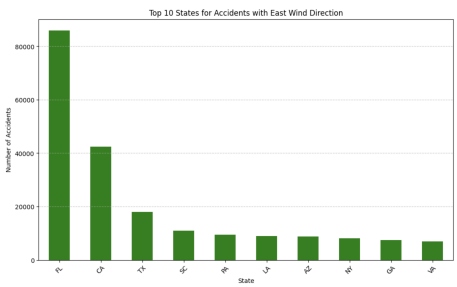


Figure 4: Left: States where most accidents happen during rain (2016-23). Right: States where most accidents happen during black-ice conditions (2016-23).

6.1.1 Weather conditions

We observed that most accidents happen during fair weather conditions when the temperatures are warm (68 F to 86 F). We speculate that this may be due to the heavy vehicle traffic that can be pretty typical for nice weather days. Also, NY tops all states for crashes during rainy days and CO tops all states for crashes during black ice conditions.



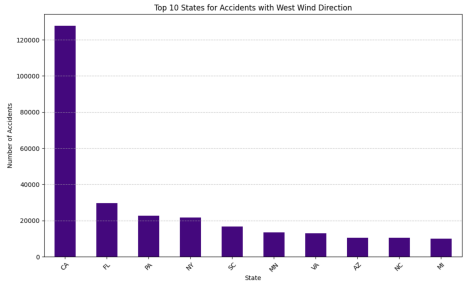


Figure 5: Left: States where most accidents happen during eastern winds (2016-23). Right: States where most accidents happen during westerly winds (2016-23).

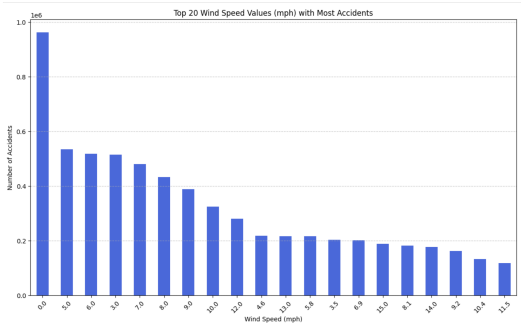


Figure 6: Wind speed during crashes(2016-23).

6.1.2 Wind conditions

Florida tops all states when it comes to having the most occurrences of eastern winds during crashes. The eastern winds are certainly plausible since Florida is on the east coast and the Atlantic could bring in the winds from the east. Similarly, California tops all states for the occurrences of Westerly winds during the accidents. Note that this doesn't necessarily mean the wind direction happened to be the primary contributor during the crashes. Also, most crashes happen during calm conditions which align with our good weather - many vehicles - many crashes conjecture.

6.1.3 Under the Influence

We also rank states by the number of occurrences of substances in the accident description column ('weed', 'substance', 'marijuana', 'blunt', 'joint', 'medication', 'drug'). California, New York and Florida top the list, whereas Colorado shows up on the list as well. The data points pertaining to these substances were very scarce, so we do not consider the dataset rich enough for us to extract useful relational insights pertaining to the influence of said substances and the crashes.

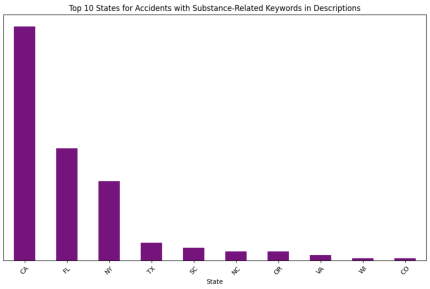


Figure 7: Ranking states based on the occurrence of a Substance in the crash description (2016-23)

6.2 Where do accidents happen?

6.2.1 Interstate Crashes

We also analyze and plot the accident heatmaps for four interstates, namely I-70, I-80, I-90 and I-25. For Interstate 70, our analysis shows the mountain corridor passing through Colorado and the urban corridor around Kansas City to be the two major accident hotspots. For Interstate 80, most accidents seem to be concentrated towards California, particularly around the Sierra Nevada Range. Intestate 90 sees a lot of accidents around Chicago, Madison NY, Albany NY and also through the treacherous mountain country around west Montana. Also, Interstate-25 sees most of its accident happen around the urban corridor passing through Denver.

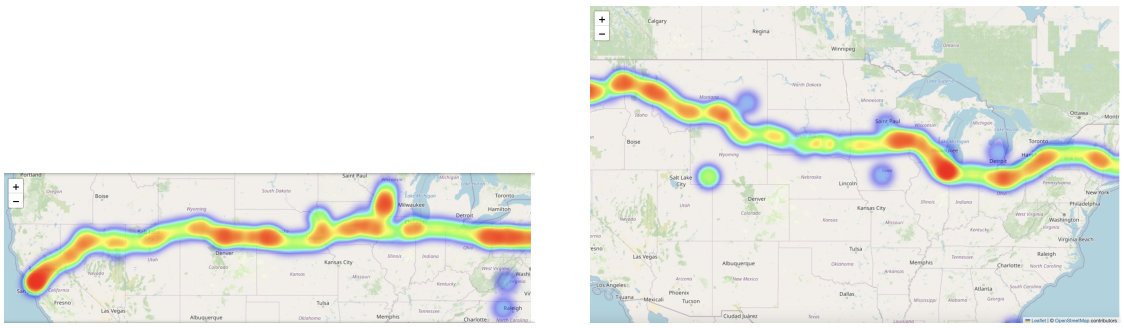


Figure 8: Left: i80 accidents Heatmap (2016-23). Right: i90 accidents heatmap (2016-23)

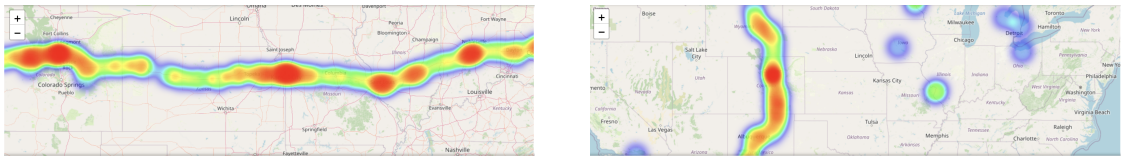


Figure 9: Left: i70 accidents Heatmap (2016-23). Right: i25 accidents heatmap (2016-23)

6.2.2 Fort Collins Crashes

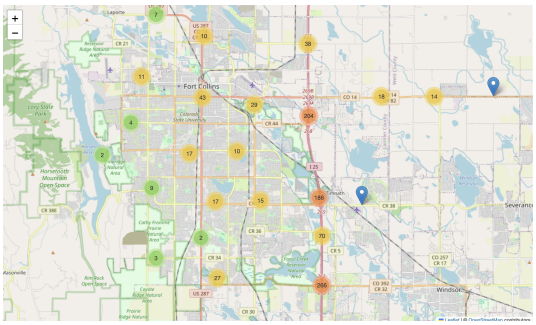


Figure 10: Fort Collins crash plot (2016-23)

6.3 Yearly Insights - US

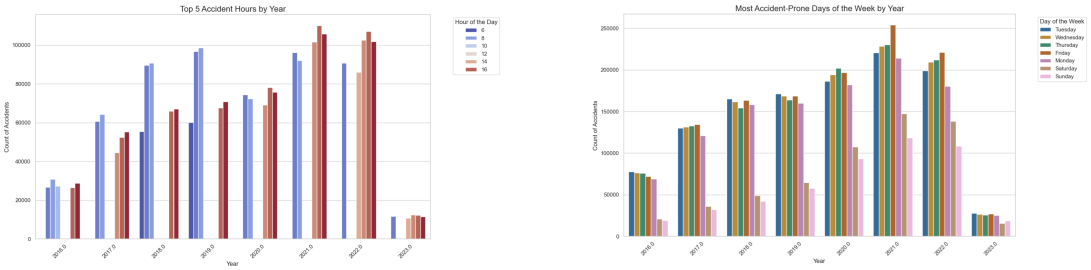


Figure 11: Hours with most accidents by year (2016-23). Right: Days of week with most accidents by year (2016-23)

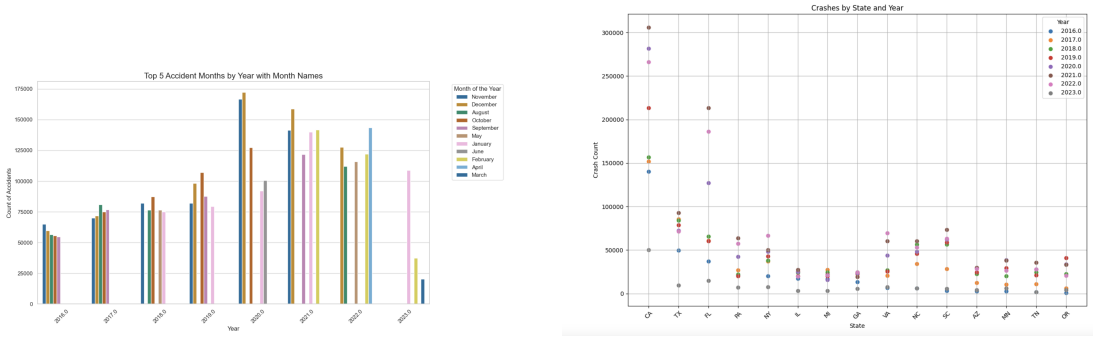


Figure 12: Months with most accidents by year (2016-23). Right: States with most accidents by year (2016-23)

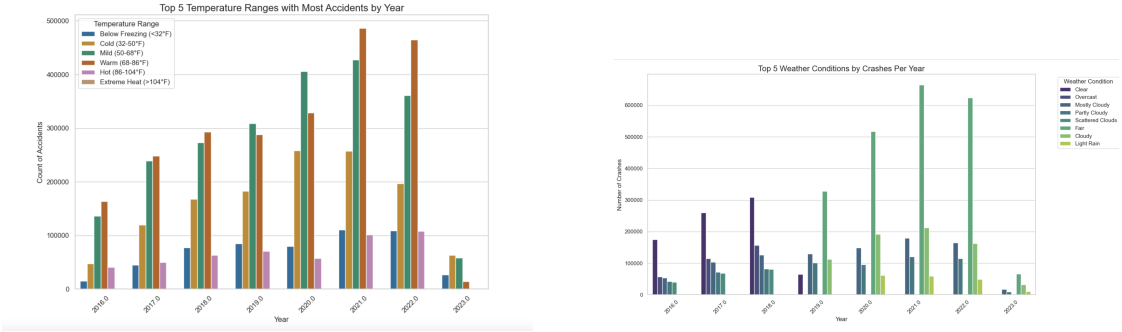


Figure 13: Temperatures during most accidents by year (2016-23). Right: Weather during most accidents per year (2016-23)

6.4 NYC - Crash Insights

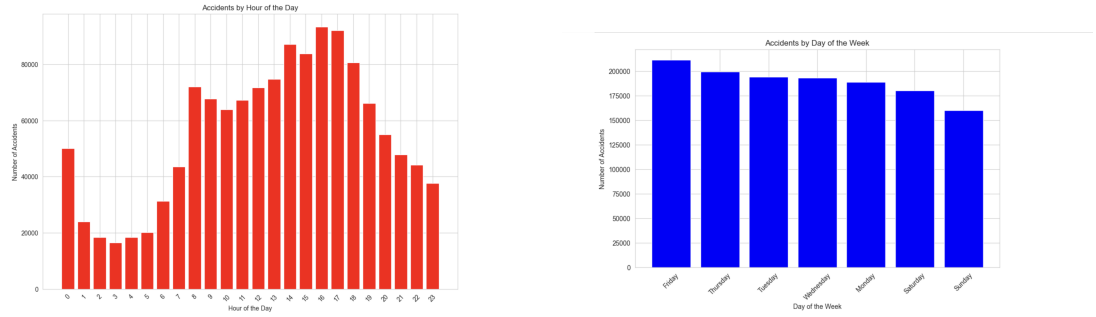


Figure 14: Hours with most accidents in NYC (2016-23). Right: Days with most accidents in NYC (2016-23)

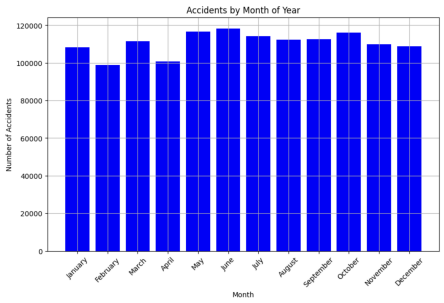


Figure 15: Months with most accidents in NYC (2016-23)

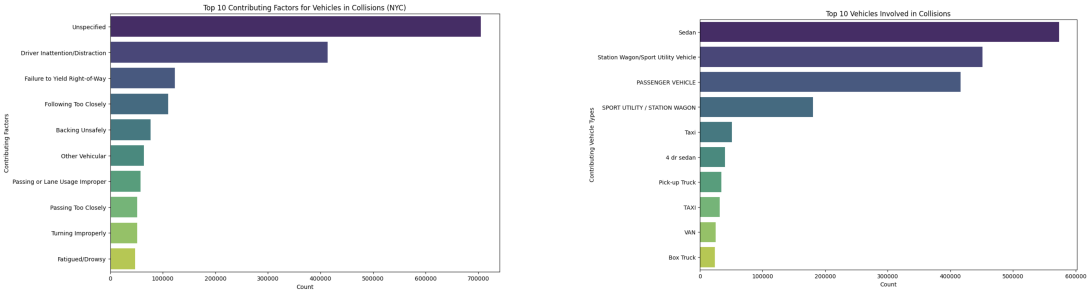


Figure 16: Causes of most accidents in NYC (2016-23). Right: Vehicle types involved in most accidents in NYC (2016-23)

6.5 Chicago - Crash Insights

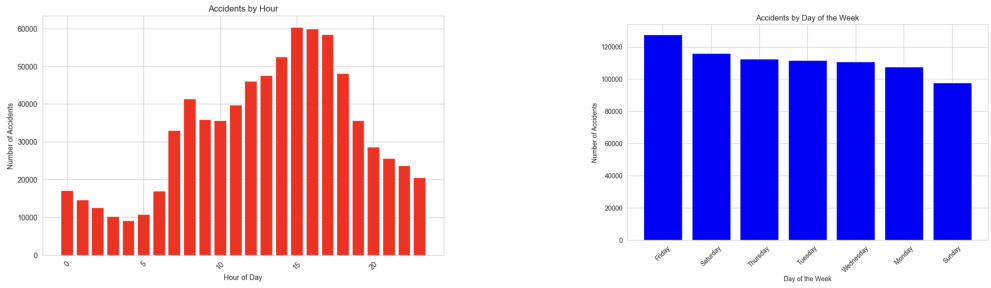


Figure 17: Hours with most accidents in Chicago (2016-23). Right: Days with most accidents in Chicago (2016-23)

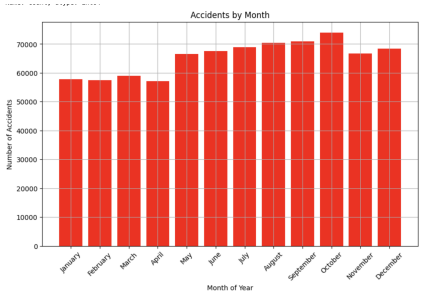


Figure 18: Months with most accidents in Chicago (2016-23)

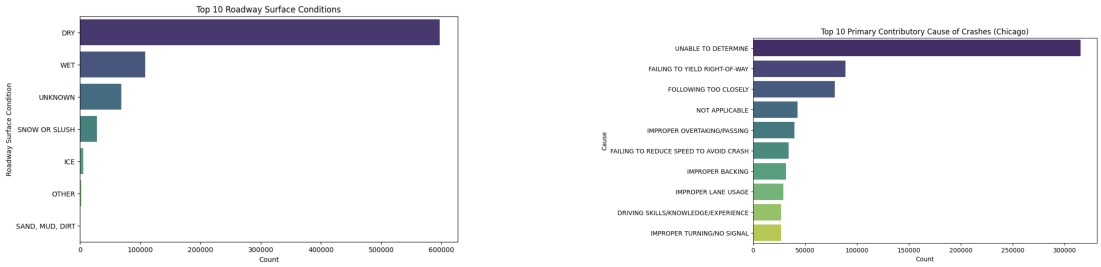


Figure 19: Road Conditions during accidents in Chicago (2016-23). Right: Causes of most accidents in Chicago (2016-23)

7 Problem space in the future

The problem we tried to solve with distributed computing framework ties into the processing of large heterogeneous traffic accident datasets and performing exploratory analytics on them. It can be noticed that the dataset we used was a culmination of contribution by various researchers that allowed us to obtain data from 2016-2023, albeit somewhat incomplete. But, researchers in the future will further contribute to this dataset or add more attributes to capture the whole picture realizing its shortcomings. There might be more other researchers that may create different other datasets that are dense and attributes aren't split across different spaces. For example, [8] have collected a traffic accidents dataset from 2016 to 2019 from the district of Setubal in Portugal and performed various machine learning approaches to analyze and predict hotspots. Other similar research from across the globe can provide newer insights and ideas to further improve upon the existing data on accidents from USA and consolidate a higher quality dataset.

In addition to that, considering the problems we faced using our current solution, benefits of distributed computing and further prospects with technological innovations, there are wide ranges of possible approaches in the future. Regarding using personal computers, with the advent of new Silicon-on-chip(SoC) computers, they could get advanced enough to outperform current computers to facilitate data analytics in personal computers. For example, Python programs ran in computer with M1 MAX chip were able to run large csv files up to certain sizes compared to traditional Intel i7 10th generation chip computer. Similarly, with new cloud solutions coming up that are getting better with each iteration, a huge move towards cloud-based big-data analytics platform is possible instead of using in-house cluster solutions[1]. This allows users to focus less on setting up the clusters and setup a data processing solution with less hassle. For example, cloud based solutions provide abstractions to users with built in applications for data storage, analytics and visualization without having to worry about setting up their own MapReduce cluster or Cassandra network. However, there are still challenges in moving data between distributed servers that are yet to be addressed [7]. But further technological improvements with tools like BigQuery, which support running complex queries and perform heavy analytical operations using parallel schemas to improve execution time[2], cloud computing can be a all round solution for storing and analyzing huge amounts of data. In gist, ease of setup and development, effective user interfaces of cloud solutions and economical factors would be huge forces that would drive distributed data processing towards cloud based solutions.

8 Conclusion

To our knowledge, our work is one of the most comprehensive analyses of the US Accidents dataset till date. Our work also differs from much of the previous work pertaining to this dataset in that we leverage distributed computation frameworks versus depending solely on a single node analytics solution like pandas. Our work leverages the power of distributed computing to speed up the analytical queries that would take significantly longer on a single node. We also fill in the gaps in the US dataset, by conducting a separate analysis of cities such as NYC and Chicago, the latter of which is missing from the US dataset. Our analysis enables glean several useful insights as described in the previous sections, and at the same time, we end up evaluating the quality of the datasets. Our observations reported several limitations of the US dataset. First, it is not as comprehensive as one may think, given

its popularity. Second, we suspect that the dataset contains a disproportionate number of sample of accident records from California and thus, is imbalanced. We have noticed California topping most of our accident categories by significant margins and thus, we see signs of data imbalance. The verification of this conjecture can be regarded as future work. Also, the black ice related accident counts for states like Wyoming, Colorado, Montana are down to almost single or double digits from 2016-23. Given our experience driving across all these states, we deem such numbers to be impossible and consider this to be an example of the scarcity of data points. There are numerous similar, impossibly small number of accidents reported for various other categories as well. Nonetheless, we have generated valuable insights from our work and we believe this will be useful to the broader research community out there.

References

[1] Mohit Agarwal and Gur Mauj Saran Srivastava. Cloud computing: A paradigm shift in the way of computing. *Journal of Cloud Computing*, 6(1):1–10, 2017.

[2] Blerim Berisha, Edlira Mëziu, and Ilir Shabani. Big data analytics in cloud computing: an overview. *Journal of Cloud Computing*, 11(1):24, 2022.

[3] Carl Boettiger. Managing larger data on a github repository. *Journal of Open Source Software*, 3(29):971, 2018.

[4] Arun Chand, S. Jayesh, and A.B. Bhasi. Road traffic accidents: An overview of data sources, analysis techniques and contributing factors. *Department of Mechanical Engineering, School of Engineering, Cochin University of Science and Technology, Kerala, India*, June 2021.

[5] Sobhan Moosavi, Mohammad Hossein Samavatian, Srinivasan Parthasarathy, and Rajiv Ramnath. A countrywide traffic accident dataset. *arXiv preprint arXiv:1906.05409*, 2019.

[6] Sobhan Moosavi, Mohammad Hossein Samavatian, Srinivasan Parthasarathy, Radu Teodorescu, and Rajiv Ramnath. Accident risk prediction based on heterogeneous sparse data: New dataset and insights. pages 33–42, November 2019.

[7] Amanpreet Kaur Sandhu. Big data with cloud computing: Discussions and challenges. *Big Data Mining and Analytics*, 5(1):32–40, 2022.

[8] D. Santos, J. Saias, P. Quaresma, and V. B. Nogueira. Machine learning approaches to traffic accident analysis and hotspot prediction. *Computers*, 10:157, 2021.

[9] World Health Organization. Road traffic injuries, 2023. Accessed: 2023-04-26.