Context-Driven Accident Severity Prediction

Group ID: G20
Members: Sashwath Krishnamoorty Giribabu, Kashyap Ava

Abstract: This project is focused on creating a context-driven prediction model for accident severity utilizing the "US Accidents" dataset[1]. Our approach involves developing ML models to predict severity based on seasonality, weather, and hotspots. The primary goal is to comprehensively understand the factors contributing to severe accidents within their contexts.

Introduction: Traffic accidents are a pervasive challenge, with significant consequences including injuries, congestion, and economic costs. Predicting accident severity is crucial for safety and management. This project's core objective is to develop context-aware prediction models for accident severity, considering accident hotspots, weather conditions, and seasonality. The demand for tailored models stems from the limitations of traditional one-size-fits-all approaches.

Motivation: We get to apply advanced techniques to a complex dataset for data cleaning, feature generation, and other methods. We get to generate different plots and derive insights from them. This project directly addresses the need for precise traffic safety measures in the real world. By creating models that consider specific contextual factors, we can provide data-driven insights for safety planning and emergency response. This project is about advancing data mining techniques while making a substantial contribution to improving traffic safety through context-aware accident severity prediction.

Related Work: Accident analysis and prediction research has evolved into three categories over recent decades. The first category examines how environmental stimuli, encompassing weather conditions and road properties, impact accident probability and severity. Researchers have delved into the influence of specific weather conditions[5,6,8], applied data mining techniques for causality analysis[7], and conducted statistical investigations to uncover the impact of unobserved variables on accident severity[4,9]. While these studies provide valuable insights into the relationships between environmental factors and accidents, their immediate practical application for real-time prediction and planning may be somewhat constrained. Second, accident frequency prediction is a prevalent focus, estimating the expected number of accidents in specific regions. Various models, including neural networks[11] and CNN[11], have been employed to predict accident frequency based on attributes such as road geometry, traffic data, and environmental factors. Lastly, the focus has been on binary classification for real-time applications[14,15]. Various models, including decision trees[13], stack-denoising autoencoders[12], and eigen-analysis[14], have been employed to anticipate the likelihood of accidents. These models have drawn on many data sources, encompassing weather conditions, traffic parameters, human behavior, and road network attributes. Their primary objective has been to distinguish pre-crash incidents from normal situations, emphasizing the significance of these variables in enhancing model accuracy. In collective efforts, these studies have enriched the traffic accident analysis and prediction field, offering an array of methodologies and insights to advance road safety and traffic management. This project builds upon and relates to existing research by incorporating context-awareness into accident severity prediction, bridging the gap between the analysis of environmental stimuli, accident frequency prediction, and severity prediction categories. Doing so aims to provide a more applicable traffic safety and management solution.

Scope: As initially proposed, the project was centered around a credit card approval prediction using vintage analysis. It was observed that the project faced under scoping and ambiguity regarding whether it should be approached as a supervised or unsupervised learning task. Importantly, the difficulty of the initially proposed project was largely contingent on the choice of vintage analysis. For these reasons, we developed a new project proposal with a good scope and a real-world application. Hence, we changed the project to Context-Driven Accident Severity Prediction.

Methodology:

Data Cleaning - The dataset contained several missing values, which enables us to use various data cleaning methodologies to prepare the dataset. We used the State and Timezone columns to create a map and used it to fill in the missing data for Timezones. We followed the same method to fill in the missing values for the Airport column. For the rest of the columns, we used the backward and forward fill with a limit of 5 since the data is sorted. We determined that 90% of the data had at least 40 non-NA values, so we dropped rows that did not have a minimum of 40 non-NA values. The columns - ['End_Lat', 'End_Lng', 'Wind_Chill(F)', 'Wind_Speed(mph)', 'Precipitation(in)'] had a large number of missing data and had less correlation with the target variable, and hence we dropped them.

Dimensionality Reduction - We have dropped columns that don't affect the model, like ['ID',' Description',' Source']. We used binning methods to simplify data complexity for machine learning models, reducing dimensionality and enhancing the interpretability of categorical columns like ['Weather', 'Pressure', 'Visibility']. Simultaneously, the conversion of datetime variables into discrete components, such as day and month, aims to systematically capture temporal patterns, aligning with our goal of developing context-aware prediction models.

Feature Generation - We created new features like Is_Rush_Hour, which is based on the hour in which the accident took place; Is_Holiday, which is based on whether the day in which the accident took place was a holiday or not; and Accident_Duration, based on existing features. We then drop the features used to create new features as they can be re-created from the new features, such as Start_Time, Date, etc.

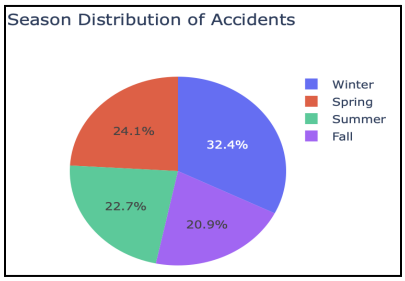
Data Preparation - Some features, like Traffic_Calming, Stop, Bump, etc., have values like "True" and "False," which were interpreted as objects. We have converted these features to a Boolean datatype. We also used Label Encoding to transform categorical features into numerical features. Some of the features where we used Label Encoding are Timezone, Wind_Direction, and City.

Feature Selection - Our strategy for predicting accident severity was rooted in considering vital contextual factors: Weather, Location, and Road Conditions. We meticulously curated features relevant to each context to tailor our predictive model effectively. For Weather, we carefully incorporated attributes such as Month, Temperature(F), Wind_Chill(F), Humidity(%), Pressure(in), Visibility(mi), Wind_Speed(mph), and Precipitation(in). Leveraging Location-related insights, we embraced Starting Longitude, Latitude, and Distance(mi) to encapsulate geographical nuances. Furthermore, our inclusion of Road Conditions encompassed a broad spectrum of variables, including Amenity, Bump, Crossing, Give_Way, Junction, No_Exit, Railway, Roundabout, Station, Stop, Traffic_Calming, Traffic_Signal, and

Turning_Loop. This holistic feature selection approach ensures our model captures and leverages the diverse contextual influences driving accident severity predictions.

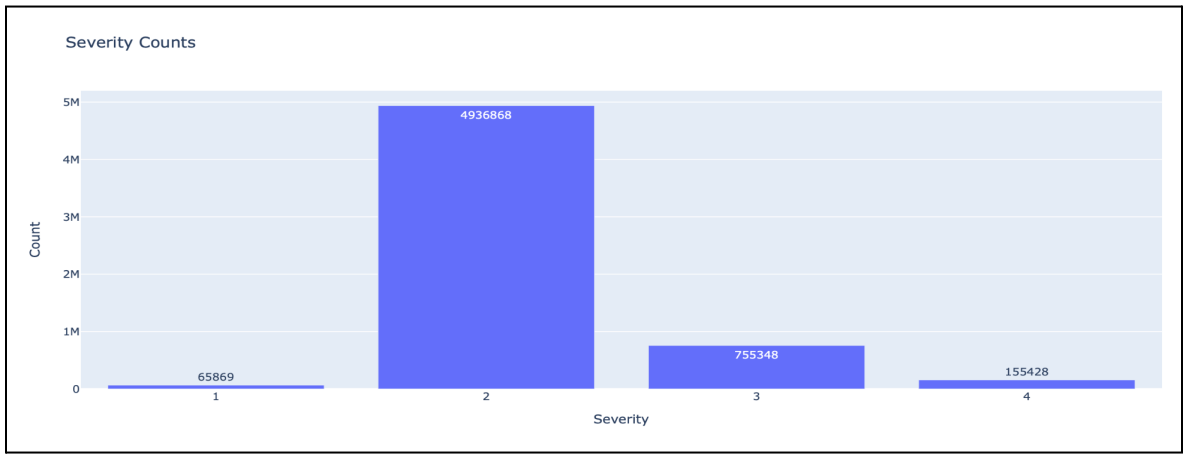
Empirical results:

Data Split - Since our project is based on predicting the severity based on the weather, we decided to split the data into two such that all the accidents that occurred during the Winter are allocated to one dataset and the rest to another. Upon further inspection, the imbalance existed in both datasets as well. So, we decided to Undersample the dataset.



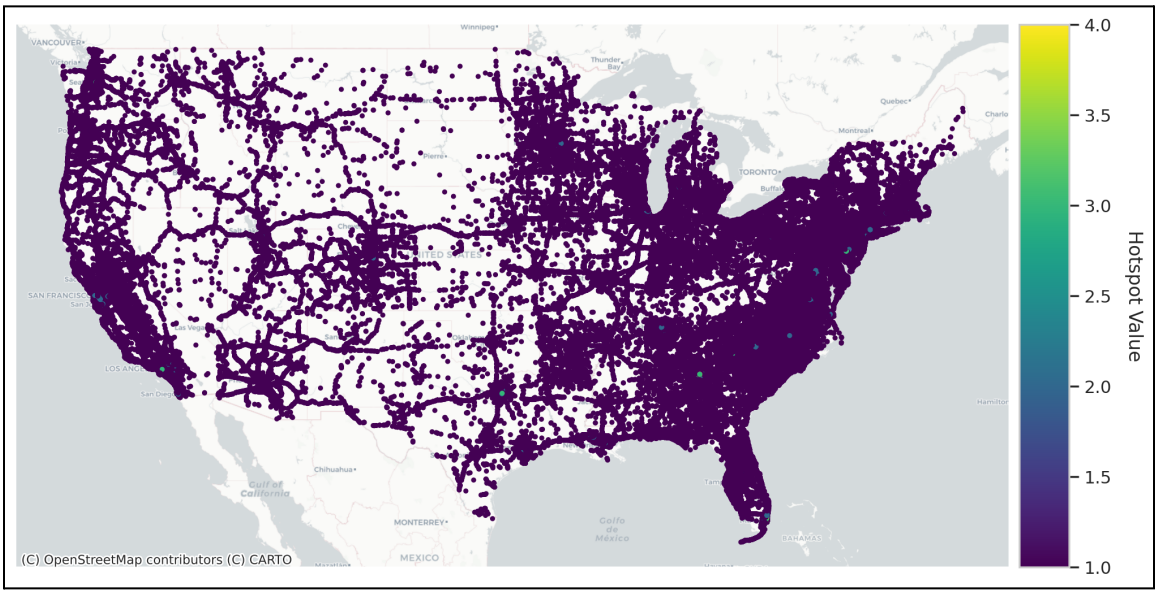
The pie chart indicates the percentage of accidents occurring in different seasons. Due to 1/3rd of the accidents occurring in a single season, the season might influence our model. Hence, we decided to develop different models for Winter vs. other Seasons.

Data Imbalance - We plotted the bar graph of the Number of Accidents vs. Severity, which indicated an imbalance in the whole dataset. The bar graph below shows that the number of data points with Severity 2 is more significant than the others. This imbalance also existed in the two separate datasets as well. This result enabled us to conclude that the data needs to be undersampled. We decided to undersample the data since oversampling would increase the dataset's size and result in computational problems. Due to the availability of various undersampling algorithms, we had to select one that would work the best for our data. So, we tried three different undersampling algorithms - RandomUnderSampler, NearMiss, and ClusterCentroids. So we first undersampled the data with each algorithm, then trained and tested the data and calculated the misclassification cost with the highest weights for misclassifying severity 4. Among these, the ClusterCentroids algorithm had the lowest cost of misclassification, thus prompting us to use this algorithm.



Hotspot Identification - The dataset is transformed into a GeoDataFrame, which adds geographical context to the information by creating point geometries based on longitude and latitude. A spatial index is designed to expedite spatial queries. The core computation examines each row in the dataset to determine the number of nearby accidents within a specified distance. This is achieved by efficiently narrowing down potential matches using spatial indexing and precisely identifying nearby accidents through spatial

intersections. The final results, representing the count of nearby accidents, are used to create a new feature indicating the level of hotspots (0-100 : 1, 100-200 : 2, 200-300 : 3, >300 : 4) integrated into the original dataset.



Model Selection - We selected the K-Nearest Neighbors and Decision Trees as our baseline model on which we trained our datasets. Following a high WSE, we pursued ensemble algorithms like Random Forest, XGBoost, AdaBoost, and Bagged Decision Trees. Due to the complex nature and large dataset size, we resorted to using a Feed-Forward Neural Network.

Model Training - We employed k-fold cross-validation, specifically setting k to 4, to meticulously divide the dataset into training, validation, and test sets using an 80-10-10 split strategy. This approach ensured a thorough evaluation of our models across multiple splits, allowing us to assess their performance effectively. Additionally, leveraging a randomized search technique, we fine-tuned the models to obtain the optimal configurations.

Evaluation Metric - We have used a tailored metric inspired by the principles of RMSE (Root Mean Squared Error) to underscore the paramount importance of precision in predicting accident severity. Our custom metric echoes RMSE's commitment to minimizing errors, with a deliberate focus on enhancing accuracy for situations where the repercussions of mispredictions carry heightened significance. This uniquely crafted metric serves as a robust yardstick, ensuring our models are adeptly equipped to address the challenges of real-world scenarios. We call it a Weighted Severity Error.

$$WSE = \frac{1}{N}\sum_{i=0}^{N} abs\,|(actual(y)_i^{\,2} - prediction(y)_i^{\,2})|$$

Results:

Winter Dataset :

Model	Parameter Grid	Best Parameter	WSE

K-Nearest Neighbour	n_neighbors: [3, 5, 10, 20] weights: ['uniform', 'distance'] p: [1, 2]	{'weights': 'uniform', 'p': 1, 'n_neighbors': 10}	171.2
Decision Tree	criterion: ['gini', 'entropy'] splitter: ['best', 'random'] max_depth: [None, 10, 20] min_samples_split: [2, 5] min_samples_leaf: [1, 2]	{'splitter': 'best', 'min_samples_split': 5, 'min_samples_leaf': 2, 'max_depth': 20, 'criterion': 'entropy'}	17.4
Random Forest	n_estimators: [50, 100, 200] criterion: ['gini', 'entropy'] max_depth: [None, 10, 20] min_samples_split: [2, 5] min_samples_leaf: [1, 2]	{'n_estimators': 200, 'min_samples_split': 2, 'min_samples_leaf': 1, 'max_depth': None, 'criterion': 'entropy', 'bootstrap': False}	7.4
XGBoost	n_estimators: [50, 100, 200] learning_rate: [0.01, 0.1, 0.2] max_depth: [3, 5, 7] subsample: [0.8, 0.9, 1.0] colsample_bytree: [0.8, 0.9, 1.0]	{'subsample': 0.8, 'n_estimators': 100, 'max_depth': 7, 'learning_rate': 0.2, 'colsample_bytree': 0.9}	12.6
AdaBoost	n_estimators: [50, 100, 200] learning_rate: [0.01, 0.1, 0.2]	{'n_estimators': 50, 'learning_rate': 0.01}	13.2
Bagged Decision Trees	n_estimators: [10, 50, 100] max_samples: [0.5, 0.7, 0.9, 1.0]	{'n_estimators': 100, 'max_samples': 0.5}	12.4
Feed-Forward Neural Network	neurons = [64, 128, 256] dropout = [0.3, 0.5, 0.7] learning_rate = [0.001, 0.01] optimizer = ['adam', 'rmsprop'] activation = ['ReLu', 'tanh']	'neurons': 256, 'dropout_rate': 0.5, 'learning_rate': 0.01, 'optimizer': 'adam', 'activation': 'ReLu'	2.82

Rest of Seasons Dataset :

Model	**Parameter Grid**	**Best Parameter**	**WSE**
K-Nearest Neighbour	n_neighbors: [3, 5, 10, 20] weights: ['uniform', 'distance'] p: [1, 2]	{'weights': 'uniform', 'p': 1, 'n_neighbors': 7}	246.06
Decision Tree	criterion: ['gini', 'entropy'] splitter: ['best', 'random'] max_depth: [None, 10, 20] min_samples_split: [2, 5] min_samples_leaf: [1, 2]	{'splitter': 'best', 'min_samples_split': 2, 'min_samples_leaf': 1, 'max_depth': 20, 'criterion': 'entropy'}	21.77
Random Forest	n_estimators: [50, 100, 200] criterion: ['gini', 'entropy'] max_depth: [None, 10, 20] min_samples_split: [2, 5]	{'n_estimators': 200, 'min_samples_split': 2, 'min_samples_leaf': 1, 'max_depth': None,	17.635

	min_samples_leaf: [1, 2]	'criterion': 'entropy', 'bootstrap': False}	
XGBoost	n_estimators: [50, 100, 200] learning_rate: [0.01, 0.1, 0.2] max_depth: [3, 5, 7] subsample: [0.8, 0.9, 1.0] colsample_bytree: [0.8, 0.9, 1.0]	{'subsample': 0.9, 'n_estimators': 100, 'max_depth': 5, 'learning_rate': 0.2, 'colsample_bytree': 1.0}	19.66
AdaBoost	n_estimators: [50, 100, 200] learning_rate: [0.01, 0.1, 0.2]	{'n_estimators': 100, 'learning_rate': 0.01}	16.835
Bagged Decision Trees	n_estimators: [10, 50, 100] max_samples: [0.5, 0.7, 0.9, 1.0]	{'n_estimators': 100, 'max_samples': 0.5}	18.355
Feed-Forward Neural Network	neurons = [64, 128, 256] dropout = [0.3, 0.5, 0.7] learning_rate = [0.001, 0.01] optimizer = ['adam', 'rmsprop'] activation = ['ReLu', 'tanh']	'neurons': 256, 'dropout_rate': 0.5, 'learning_rate': 0.01, 'optimizer': 'adam', 'activation': 'ReLu'	1.7

Conclusion:

In conclusion, this project successfully addressed the challenge of predicting accident severity using a context-aware approach on the "US Accidents" dataset. Through diligent data cleaning, effective undersampling, and exploring diverse machine learning models, Feed-Forward Neural Network emerged as the most effective algorithm, particularly when tailored for different seasons. The decision to develop separate models for winter and other seasons proved instrumental in capturing nuanced patterns. This work contributes to advancing traffic safety measures and underscores the importance of context-specific modeling. Future directions include refining feature engineering, implementing dynamic model training, and exploring real-time applications, paving the way for continued progress in traffic safety through data-driven methodologies.

This domain holds promising opportunities for further exploration and research in the foreseeable future. Further enhancement of feature engineering could delve into intricate contextual details, providing a more nuanced understanding of accident severity predictors. Dynamic model training mechanisms can be explored to adapt models to evolving traffic dynamics and external influences. Real-time implementation presents an exciting prospect for immediate emergency response and traffic management applications. Additionally, integrating the developed models with existing traffic systems could yield proactive safety measures. These directions propel the field forward, ensuring that data-driven methodologies remain at the forefront of improving traffic safety, responding to emerging challenges, and fostering more resilient and adaptive systems.

References:

1. Sobhan Moosavi. (2023). US Accidents (2016 - 2023). Retrieved from https://www.kaggle.com/datasets/sobhanmoosavi/us-accidents.
2. Moosavi, Sobhan, Mohammad Hossein Samavatian, Srinivasan Parthasarathy, and Rajiv Ramnath. "A Countrywide Traffic Accident Dataset.", 2019.
3. Moosavi, Sobhan, Mohammad Hossein Samavatian, Srinivasan Parthasarathy, Radu Teodorescu, and Rajiv Ramnath. "Accident Risk Prediction based on Heterogeneous Sparse Data: New Dataset and Insights." In Proceedings of the 27th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, ACM, 2019.
4. Joaquín Abellán, Griselda López, and Juan De OñA. 2013. Analysis of traffic accident severity using decision rules via decision trees. Expert Systems with Applications 40, 15 (2013), 6047–6054
5. Daniel Eisenberg. 2004. The mixed effects of precipitation on traffic crashes. Accident analysis & prevention 36, 4 (2004), 637–647.
6. David Jaroszweski and Tom McNamara. 2014. The influence of rainfall on road accidents in urban areas: A weather radar approach. Travel behavior and society 1, 1 (2014), 15–21.
7. Sachin Kumar and Durga Toshniwal. 2015. A data mining framework to analyze road accident data. Journal of Big Data 2, 1 (2015), 26.
8. JD Tamerius, X Zhou, R Mantilla, and T Greenfield-Huitt. 2016. Precipitation effects on motor vehicle crashes vary by space, time, and environmental conditions. Weather, Climate, and Society 8, 4 (2016), 399–407.
9. Athanasios Theofilatos. 2017. Incorporating real-time traffic and weather data to explore road accident likelihood and severity in urban arterials. Journal of Safety Research 61 (2017), 9–21.
10. Li-Yen Chang. 2005. Analysis of freeway accident frequencies: negative binomial regression versus artificial neural network. Safety Science 43, 8 (2005), 541–557
11. Ciro Caliendo, Maurizio Guida, and Alessandra Parisi. 2007. A crash-prediction model for multilane roads. Accident Analysis & Prevention 39, 4 (2007), 657–670.
12. Quanjun Chen, Xuan Song, Harutoshi Yamada, and Ryosuke Shibasaki. 2016. Learning deep representation from big and heterogeneous data for traffic accident inference. In Thirtieth AAAI Conference on Artificial Intelligence. AAAI, Palo Alto, CA, USA.
13. Lei Lin, Qian Wang, and Adel W Sadek. 2015. A novel variable selection method based on frequent pattern trees for real-time traffic accident risk prediction. Transportation Research Part C: Emerging Technologies 55 (2015), 444–459.
14. Zhuoning Yuan, Xun Zhou, Tianbao Yang, James Tamerius, and Ricardo Mantilla. 2017. Predicting traffic accidents through heterogeneous urban data: A case study. In Proceedings of the 6th International Workshop on Urban Computing (UrbComp 2017), Halifax, NS, Canada, Vol. 14. ACM, New York, NY, USA.
15. Lu Wenqi, Luo Dongyu, and Yan Menghua. 2017. A model of traffic accident prediction based on convolutional neural network. In 2017, the 2nd IEEE International Conference on Intelligent Transportation Engineering (ICITE), IEEE, 198–202.