Crash Map: Identifying Severe Crash Areas

Hunter Apple, Michael Barnhart, Kyle Kaveny,
Kunal Shah, Alexei Vinogradov, Yangpeiyun Xu

Introduction

In recent years, there has been an increasing initiative to leverage technology to keep roads safer and reduce traffic accidents, which are one of the leading causes of death in the US [1]. While there is great progress being made in car technology, such as autonomous driving systems, driver assistance systems [2], lane departure warnings, and blind-spot detection, we see an opportunity for improvement in map routing software [3]. We aim to inform drivers of severe crash locations due to adverse weather conditions. We created Crash Map, an interactive mapping tool that estimates the severity of an automobile accident, given a specific location, time of day, and a set of weather conditions.

Problem Definition

Despite advances in driving technology, car accidents are still a major cause of injury and death. Current technology, such as GPS and mapping software, provide a driver with a route to their destination and is optimized for time (i.e., finding the fastest route) or optimized for reduced carbon footprint. While this is useful from a logistical and an ecological standpoint, we seized an opportunity to optimize for safety and put human life first. We addressed the problem by creating an interactive tool that informs drivers about route safety based on parameters such as weather, time of day, and location.

Survey

The U.S. DOT reports that 24% of automobile crashes are due to weather related conditions, resulting in 7,400 deaths and 673,000 injuries each year [4]. Every year adverse weather contributes to about 1 billion hours spent in traffic and accounts for around 25% of traffic delays [5]. Weather-related conditions that contribute to crashes include snow, rain, fog, ice, sleet, or wind. [6],[7] These conditions affect road safety by the reduction of pavement friction and road visibility [8]. Research shows that machine learning (ML) techniques such as decision trees, random forests (RF), and DNN's can produce accurate and significant results when working with crash data [9]. We believe that our interactive mapping tool has the potential to reduce the number of weather-related crashes that occur by rerouting or warning drivers of the potential crash severity due to poor weather. This information would benefit drivers with GPS, logistics-based companies (e.g., UPS, Amazon), and government institutions such as the DOT and NHTSA [10].

There is a considerable amount of research illustrating how various factors affect traffic and crash severity such as weather and time of day variations in driver performance [11-13]. A good starting point for our project was from a paper that uses "pattern discovery" in traffic and weather data to find patterns of collocation, co-occurrence, cascading, or cause and effect between geospatial entities [14]. This paper shows different factors that create crashes and traffic congestion. Additional research, focusing on the prediction of crash risk using ML models such as RF and LSTM-CNN, gave us benchmark results and a starting point for the models we will be generating from our datasets. Unlike many studies in this field, which focus only on freeways, the LSTM-CNN model focused on urban arterials to predict real-time crash risk. The RF was created with the same dataset we will be using and presented baseline visualizations that we can build upon [15],[16].

Generally, the current methods for GPS rerouting are concerned with finding the optimal route to

reduce drive time [17]. While there are instances when rerouting for optimal drive time will guide the user to avoid poor weather conditions, the main consideration of the current technology is to provide the driver with the fastest route. Weather effects can significantly reduce the capacity of road networks and driving speed [18],[19]. These factors may inadvertently lead GPS systems to find a route that has better weather conditions because the system may have been simply avoiding a congested area. Current GPS routing technology puts the responsibility on the user to check for weather conditions before driving, but these same systems have the potential to unwittingly navigate around poor weather conditions and areas that could cause severe accidents. The limitation of current technology is that GPS routing does not proactively avoid high-risk routes and has to wait for roads to become congested from weather conditions before suggesting alternate routes.

Proposed Method

Considering past research on weather's effect on crashes [4],[5],[8],[11],[12] and past research in using ML models with crash data [9],[15],[16],[20], our intuition was that training an ML model using historical weather, time, and crash location data could accurately predict crash severity. Additionally, we determined there is an unaddressed need to inform drivers about route safety since most mapping and navigation softwares' main goal is reducing travel time. Our intuition was that an interactive map that highlighted historical crash locations based on severity would allow drivers to make informed routing decisions. Crash Map informs drivers about route safety using a model trained on historical crash data, emphasizing driver safety, unlike the current state of the art tools, which only focus on route time optimization .

We explored a variety of ML techniques, such as Random Forest, Extremely Random Forest, Gradient Boosting Classifier, and XGBoost, to find the best approach for predicting the severity of traffic crashes given weather conditions, time, and location. Our innovation was to leverage a combination of historic weather and accident data to determine the severity of a potential traffic accident. Using the estimated severity, we utilized D3 and Leaflet to create a first-of-its-kind interactive tool that considers the time of day, weather, and location to highlight areas of potentially severe accidents for a driver. We believe the implementation of our tool was successful because we had access to data on historical accidents and because prior research shows that ML techniques can be used to predict crash severity. In its current state, we believe our tool has the potential to improve a driver's ability to stay safe on the road.

We used the *US Accidents (4.2 million records)* dataset [1] which contains car crash data and their severities from 2016-2020. We experimented with data from both California and New York, due to the high incidence of crashes in these states, but we chose New York because of its significant amount of data (190,000 accident records), a good mix between urban and rural environments, and wide variety of weather conditions. Crash Map could be expanded to other areas and cities as more traffic data is collected and made available. Our dataset contains 129 different weather conditions, but we grouped them into 6 different categories: sunny, cloudy, fog, rain, thunderstorm, and snow or ice to reduce complexity. Our map is informational and allows users to see the potential crash severity throughout the state due to weather, specific location, time of day, and road type. Our goal is to empower drivers to make safer routing decisions with the information from our mapping tool.

We determined that in order to preserve the scale of location data for making predictions, we should utilize models that do not require normalization, such as tree-based models. Therefore, we investigated Random Forest, Extremely Random Forest, Gradient Boosting Classifier, and XGBoost. Boosting is a technique where new models are added to reduce errors made by existing models, and Gradient Boosting uses gradient descent to minimize the loss function.

XGBoost is an implementation of gradient-boosted decision trees designed for speed and performance. After we evaluated our models for estimating crash severity, we integrated the most accurate model with D3 and Leaflet to visualize the crash severity throughout the map area. D3 allows for personalized customization and interactivity, giving the user control over a variety of factors such as weather, time of day, and day of week, while Leaflet provides a navigation map as the base layer and data points as feature layers.

While many mapping software options display the recommended route, our mapping tool identifies areas of potentially high crash severity along the recommended route. This provides the driver with vital information to be cautious in specific areas, and allows the driver to decide if that route is worth taking. Our mapping tool utilizes D3 for our interactive map, Leaflet for routing and navigation, and Flask to serve the model results for our visualization. XGBoost is used to label the different severities for locations based on historical crash data. Our interface collects user input on trip start and end locations, as well as time and weather conditions, and produces turn-by-turn directions, and highlights the route on the map. The inputted weather data are subsequently used to predict either high or low severity locations along their route. Our tool also gives routing information so that you can see what roads you will be driving on, and what streets to turn at. This can be used by drivers to identify early on which roads to avoid.

We believe our approach offers a valuable supplement to traditional mapping software because historical crash data, and factors contributing to crashes, are not explicitly used by any mapping applications. We believe these factors, such as weather, can help to provide route safety information to drivers. Some of the innovations in our project include the use of weather effects to inform drivers of the potential crash severity of their routes, the D3 and Leaflet framework that allows drivers to select factors to consider when creating a route, the use of XGBoost to predict crash severity, and the empowerment of drivers to make informed routing decisions without having to listen to driving and weather reports.

Experiments and Evaluation
Data Analysis:
Our dataset contains accident data from 2016-2020 including the location, time of day, weather conditions, road type and point-of-interest features, day or night designation, descriptions, and the severity of the crash. During data cleaning, we removed extraneous information that was not useful for model creation, such as the name of the street, and any features that had large amounts of missing data. Missing weather stats were replaced with 0s, and any remaining rows left with missing data were dropped. Time was grouped by hour of the accident. Microsoft Azure was leveraged for this task due to the size of the initial dataset (1.6 GB). The features left were: Severity, Start_Lat, Start_Lng, Temperature(F), Humidity(%), Wind_Speed(mph), Precipitation(in), hour, day_of_week, Weather_Condition, and Sunrise_Sunset.

Our goal was to create a model that predicted the severity of a car accident based on factors around the accident. Rather than predicting the likelihood of an accident, which requires knowing the volume of cars passing the same point that did not result in an accident, we focused on the severity of an accident, making the assumption that the accident would happen regardless of contributing factors. Predicting the likelihood of an accident is currently out of the scope of this project due to difficulty of obtaining traffic volume data.

To predict crash severity, we decided to use ensemble-based models that did not require normalization to preserve the scale of location data, such as Random Forest or XGBoost. We tested the performance of each model against a subset of data to validate our models and a separate test set to test the model that has the highest validation accuracy, and generalizes well

to unseen data. During operation of Crash Map, the predictions are sent to the front end page (i.e., the map) to show the estimated severity of crashes in the area.

We tested four ensemble-based methods to determine their performance at predicting crash severity. The four models tested were: Random Forest, Extremely Random Forest, Gradient Boosting Classifier, and XGBoost. Each model was trained on the same training subset of the data (60% of the data) and GridSearchCV from scikit-learn was used to tune for the best parameters. Next, the models were tested on the same validation set (20% of the data) to determine the best performing model. We found that XGBoost provided the most accurate model with an overall accuracy of 87%, see Table 1, and precision of 88% and 85% for Low and High Crash Severity respectively. Gradient Boosting was the second best model with an overall accuracy of 83% and precision of 85% and 79% for Low and High. We found that the random tree models performed equally the worse with an accuracy of 64% by just labeling everything Low for the validation set and getting that accuracy by default. We tested XGBoost against a final test set (20% of the data) to determine real world performance and found the model has an accuracy of 96% and precision of 96% and 97% for Low and High. XGBoost became our model of choice for the prediction of crash severity for the application.

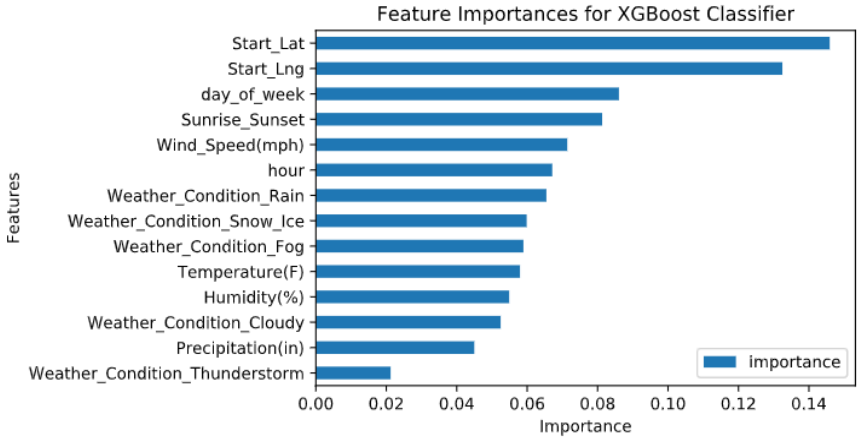
Table 1

Metrics Across Classifiers

Classifier	Label	Precision	Recall	f1-score	Accuracy
Random Forest	Low	0.64	1.00	0.78	0.64
	High	0.00	0.00	0.00	
Extremely Randomized Trees	Low	0.64	1.00	0.78	0.64
	High	0.00	0.00	0.00	
Gradient Boosting	Low	0.85	0.90	0.87	0.83
	High	0.79	0.72	0.75	
XGBoost	**Low**	**0.88**	**0.92**	**0.90**	**0.87**
	High	**0.85**	**0.78**	**0.81**	

This Table shows the classification metrics for each model based on the predictions each model outputs using a validation set of data. Each number can be seen as a percent.

Table 2



This chart shows the relative importance of each feature in the XGBoost model. The higher the importance, the more influential the feature is for severity prediction

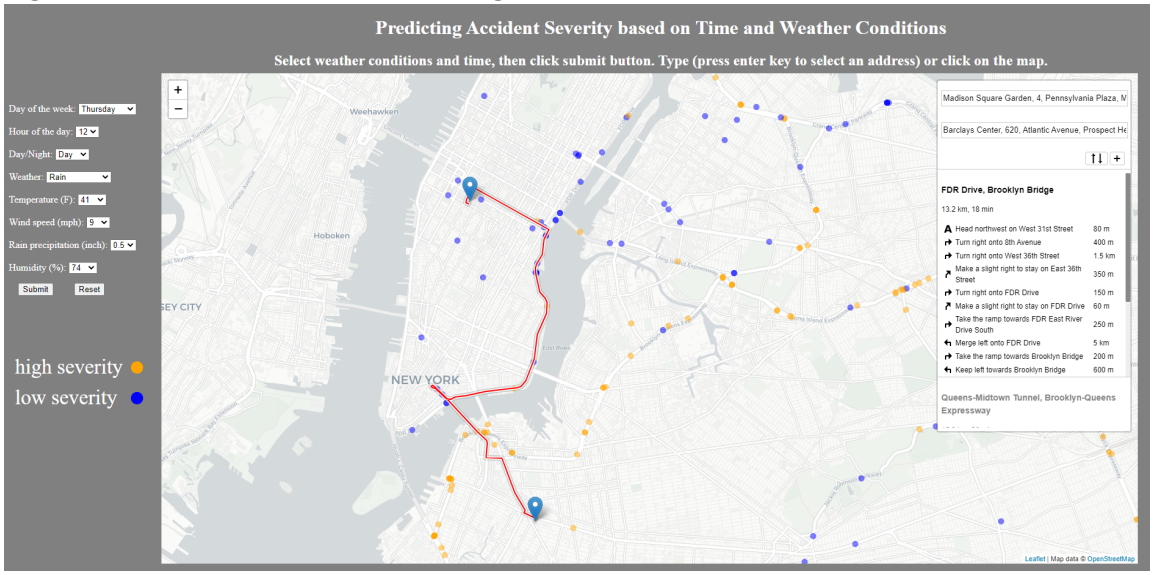
Because XGBoost is a tree-based modeling technique, we can gain insight into which features contributed the most in the prediction of crash severity. Looking at Table 2, we can see that

Starting Latitude and Longitude (Start_Lat and Start_Lng) have the biggest impact on the prediction of severity. This implies that the location of the accident has the largest impact on the severity, which follows the intuition that having an accident at an intersection will likely result in a higher severity accident compared to one occurring on a country road. Other important factors include day of the week, which correlates to the number of people driving, and Sunrise/Sunset (day or night designation) which influence road visibility.

Visualization:
Our goal was to present the severities on a navigation map given selected time and weather conditions. The visualization part of our project leverages two tools: D3 and Leaflet. D3 is a JavaScript library that generates dynamic and interactive data visualizations in web browsers. Leaflet is an open-source JavaScript library for interactive maps. Combining the two tools, we were able to plot the predicted severity values of certain locations in New York State. Leaflet provided the framework to create a navigation map as a base layer and allows for the addition of feature layers (data points) on the top of a navigation base map. It also provides basic map functions, such as zooming in, zooming out, and dragging. Routing was achieved by using an external plugin, Leaflet Routing Machine, which provides a number of routing functions, such as route search, displaying itinerary, reversing routes, and adding stops.

We created a user-friendly web application that helps drivers make safer routing decisions. We set eight dropdown boxes to let the user select weather conditions and time information (Fig. 1). Day of the week (Monday to Sunday), time of the day (0 to 23), and day/night allows users to choose the time of their trip. Users also need to select weather, temperature, wind speed, rain precipitation and humidity of the day. Weather has six options: sunny, cloudy, fog, rain, thunderstorm, and snow or ice as discussed in the Data Analysis section. We kept the ranges and the units of temperature, wind speed, rain precipitation and humidity the same as the ranges of these variables in the accident dataset. When the user clicks the submit button, the predicted severities show up on the map, with orange points indicating high severities and blue points indicating low severities. Users are able to zoom in to see the accident hot spots along their routes. The reset button will clear the user input, the severities on the map, and the route so that the user can start a new selection.

Figure 1: A snapshot of the mapping tool



In Python, the accidents dataset and XGBoost model are loaded. After a user submits their selected parameters, Flask, a web framework python package, receives the selections and stores the data. The Python code then randomly samples 2500 data points from New York and passes them to the XGBoost model along with the parameters selected by the user. The reason for showing only 2500 data points was to ensure the user interface did not get bogged down with rendering hundreds of thousands of points. After predicting the severities for the selected data points, the Flask framework passed the data back to D3 to be displayed for the user.

During development, the code (D3 and Python), dataset, and model were hosted in a GitHub repository. After completing the development, this repository was pushed to Heroku, which is a cloud application platform. Crash Map is hosted at: https://crashmap-api-heroku.herokuapp.com/ and can be accessed from Android phones, tablets, or computers.

We also tried to display the severities only for the area that the user is interested in (i.e., near the route). We were able to reduce the number of the crash locations displayed based on the start location and the end location of the route. But we were having trouble updating the points after the route was calculated since Leaflet does not allow adding additional data layers after route search. We also tried to limit the points to those only on the route, but we were unable to extract the latitude and longitude of the points along the route. This feature was left as a potential future direction.

Conclusions and Discussion

The original goal of Crash Map was to predict the likelihood and severity of a crash along a given route. Over the course of the development of Crash Map we found predicting crash likelihoods would require traffic information for the volume of cars passing a given location. This led us to pivot to predicting the crash severity, which would inform drivers of historically dangerous routes. We identified XGBoost as our best performing model with a validation accuracy of 87% and a test accuracy of 96% for predicting severity. These predicted severities were then displayed by D3 and Leaflet to show the areas of high and low severity on a map. Crash Map prompts the user to give time, weather, and location information that will be used to predict the severities along the user's route.

We believe Crash Map has the potential to save lives. It fills a gap presented by popular mapping and navigation software, which is the lack of proactive avoidance of routes adversely affected by weather conditions. While we have built a standalone tool that provides both routing and safety information, we also see an opportunity to integrate our logic and models with these popular mapping software. The goal is that users have safety information provided to them in real-time while planning or driving their route.

Distribution of Team Member Effort

All team members have contributed a similar amount of effort.

References

1. Moosavi, S., Saavatian, M. H., Parthasarathy, S., & Ramnath, R. (2019). A Countrywide Traffic Accident Dataset.", arXiv preprint arXiv:1906.05409.

2. Xue, Q., Wang, K., Lu, J. J., & Liu, Y. (2019). Rapid driving style recognition in car-following using machine learning and vehicle trajectory data. Journal of Advanced Transportation, 2019, 1-11. doi:10.1155/2019/9085238

3. Freitas, T. R., Coelho, A., & Rossetti, R. J. (2010). Correcting routing information through gps data processing. 13th International IEEE Conference on Intelligent Transportation Systems. doi:10.1109/itsc.2010.5624996

4. Pisano, P. A., Goodwin, L. C., & Rossetti, M. A. (2008). U.S. Highway Crashes in Adverse Road Weather Conditions.

5. Alfelor, R. M., & Yang, C.Y. D. (2011). Managing Traffic Operations During Adverse Weather Events.

6. Eisenberg, D. (2004). The mixed effects of precipitation on traffic crashes. In Accident Analysis & Prevention, 36(4), 637-647. doi:10.1016/s0001-4575(03)00085-x

7. Qiu, L., & Nixon, W. A. (2008). Effects of adverse weather on traffic crashes. Transportation Research Record: Journal of the Transportation Research Board, 2055(1), 139-146. doi:10.3141/2055-16

8. Khan, G., Qin, X., & Noyce, D. A. (2008). Spatial analysis of Weather Crash Patterns. Journal of Transportation Engineering, 134(5), 191-202. doi:10.1061/(asce)0733-947x(2008)134:5(191)

9. Yuan, Z., Zhou, X., Yang, T., Tamerius, J., & Mantilla, R. (2017). Predicting Traffic Accidents Through Heterogeneous Urban Data: A Case Study. In Proceedings of 6th International Workshop on Urban Computing, Halifax, Nova Scotia, Canada, August 2017.

10. Litzinger, P., Navratil, G., Sivertun, Å, & Knorr, D. (2012). Using weather information to improve route planning. Lecture Notes in Geoinformation and Cartography, 199-214. doi:10.1007/978-3-642-29063-3_11

11. Howard, B., Parshall, L., Thompson, J., Hammer, S., Dickinson, J., & Modi, V. (2012). Spatial distribution of urban building energy consumption by end use. Energy and Buildings, 45, 141-151. doi:10.1016/j.enbuild.2011.10.061

12. Lenné, M. G., Triggs, T. J., & Redman, J. R. (1997). Time of day variations in driving performance. Accident Analysis & Prevention, 29(4), 431-437.

13. De Cauwer, C., Verbeke, W., Coosemans, T., Faid, S., & Van Mierlo, J. (2017). A data-driven method for energy consumption prediction and energy-efficient routing of electric vehicles in real-world conditions. Energies, 10(5), 608. doi:10.3390/en10050608

14. Moosavi, S., Samavatian, M. H., Nandi, A., Parthasarathy, S., & Ramnath, R. (2019). Short and Long-term Pattern Discovery Over Large-Scale Geo-Spatiotemporal Data. In proceedings of the 27th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, ACM.

15. Li, P., Abdel-Aty, M., & Yuan, J. (2020). Real-time crash risk prediction on arterials based on LSTM-CNN. Accident Analysis & Prevention, 135, 105371. doi:10.1016/j.aap.2019.105371

16. Parra, C., Ponce, C., & Rodrigo, S. F. (2020). Evaluating the performance of Explainable machine learning models in traffic Accidents prediction in California. 2020 39th International Conference of the Chilean Computer Science Society (SCCC). doi:10.1109/sccc51225.2020.9281196

17. Thai, J., Laurent-Brouty, N., & Bayen, A. M. (2016). Negative externalities of gps-enabled routing applications: A game theoretical approach. 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC). doi:10.1109/itsc.2016.7795614

18. Sathiaraj, D., Punkasem, T., Wang, F., & Seedah, D. P. (2018). Data-driven analysis on the effects of extreme weather elements on traffic volume in Atlanta, GA, USA. In Computers, Environment and Urban Systems, 72, 212-220. doi:10.1016/j.compenvurbsys.2018.06.012

19. Maze, T. H., Agarwal, M., & Burchett, G. (2006). Whether weather matters to traffic Demand, traffic safety, and Traffic operations and flow. Transportation Research Record: Journal of the Transportation Research Board, 1948(1), 170-176. doi:10.1177/0361198106194800119

20. Moosavi, S., Hossein, M., Parthasarathy, S., Teodorescu, R., & Ramnath, R. (2019). Accident Risk Prediction based on Heterogeneous Sparse Data: New Dataset and Insights. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. ACM.