From Data to Action: Predicting Traffic Accident Severity for Emergency Response

Nathen Hale Fernandes, Alexander Armstrong, Marc Crasto

Abstract

Machine learning applications in public safety offer data-driven insights that enhance emergency response and urban planning. This project leverages the Seattle Traffic Collision Dataset to predict traffic accident severity - property damage, injury, or severe, using features like weather, road surface, and lighting conditions, etc. We tested Decision Trees, Random Forests, and Gradient Boosting models. These findings could improve resource allocation and reduce emergency response times, contributing to smarter and safer urban systems.

Background

The rise of smart cities has fueled interest in leveraging data science to enhance public safety. Predicting traffic accident severity is critical for prioritizing emergency resources and identifying high-risk areas. While traditional studies focused on accident frequency or location, the analysis of severity remains less explored but equally impactful for emergency response and urban planning.

Previous research highlights environmental and situational factors such as weather, road conditions, and time of day as key predictors of accident severity.[5, 7, 11] For example, studies in the UK (2019) and UAE (2020) demonstrated the effectiveness of Decision Trees and Gradient Boosting models, respectively, in predicting severity, emphasizing the importance of features like weather and lighting conditions.[1, 2] Another study in the US (2022) showed Random Forest models optimized with Bayesian techniques to be effective but revealed challenges in predicting severe accidents due to imbalanced data[4].

Our work builds on these findings using the Seattle Traffic Collision Dataset, which offers detailed records from 2004–2020, including factors like weather, road surface, and lighting. By refining feature selection and addressing class imbalance, our study aims to advance severity prediction models to support emergency response systems.

Data Preprocessing

For our model, we use various features from the Seattle Traffic Collision dataset to predict accident severity. Key features include weather conditions (e.g., clear, rain, snow), road surface conditions (e.g., dry, wet, icy), and lighting conditions (e.g., daylight, dark with street lights, etc.). The target variable for our model is the severity code, which classifies accidents by severity level (e.g., unknown (0), property damage (1), injury (2), serious injury (2b), fatal (3)). Our data preprocessing pipeline consisted of three key stages: feature creation, data cleaning, and feature selection.

Before proceeding to feature creation, we made some preliminary modifications to the 'Collisions.csv' dataset. Values of '2b' in the SEVERITYCODE column were replaced with 3 (consolidating the two classes into one for better classification), a step typically done during data cleaning, but adjustments were made earlier due to encountered errors. Several categorical variables were converted to numerical values, and NaN entries in specific categorical columns were replaced with 0. The updated dataset was then saved as 'Modified_Collisions.csv' to resolve errors that arose when working directly with the original file.

In feature creation, we developed High-Risk Driver and Environmental Risk features to enhance predictive accuracy. The High-Risk Driver score integrates data on driver distractions (INATTENTIONIND) or intoxicated (UNDERINFL). Another feature that we came up with is the Environmental Risk score which accounts for weather (WEATHER) and visibility conditions

(LIGHTCOND). Recognizing the impact of temporal patterns on traffic accidents, we also introduced two new features: Season (categorizing incidents into Spring, Summer, Fall, or Winter) and Weekday (labeling days from Sunday to Saturday). These features provided valuable contextual information, significantly improving the models' predictive performance.

During data cleaning, rows with unknown severity (class 0) were removed. Columns with over 10% missing values, non-predictive attributes, and those potentially leaking target information, such as the number of fatalities, were dropped. Categorical variables (such as WEATHER, ROADCOND, and LIGHTCOND) were processed using one-hot encoding and the rest of the categorical variables were processed with Label Encoding but not before dropping rows with null attributes. This approach ensures minimal information loss while maintaining data consistency, drawing on methods highlighted in Elyassami et al.'s study[1]. We also label encoded the target variable (so that it can be processed by the mutual information classifier in the feature selection process), so now the labels are 0 (property damage), 1 (injury), and 2 (severe).

Last stage of the data processing is Feature Selection. For feature selection, we used a two-phase approach: This involves using a mutual information classifier to rank features by their dependency on the target. Then we pass on a subset of these features into the second phase, the Recursive Feature Elimination with Cross Validation [5]. This removes less important features and optimizes model performance. This process improved accuracy and reduced overfitting by focusing on the most informative features. Through feature creation, data cleaning, and feature selection, we refined the dataset to build models that effectively predict accident severity and support emergency response decisions.

Visualisation

To analyze whether the number of accidents varies with seasonality and the day of the week, heatmaps were created to explore these relationships.

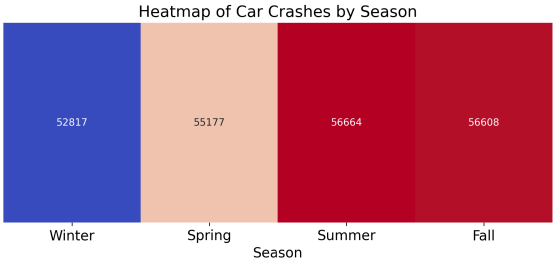

Figure 1a. This figure shows the heatmap for the number of accidents during each season

Figure 1a shows accidents peak in summer, suggesting seasonal patterns[11, 12].

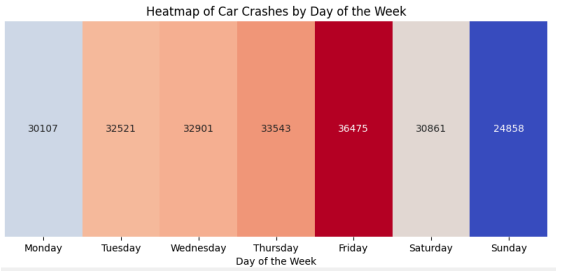

Figure 1b. This figure shows the heatmap for the number of accidents for each day of the week

The heatmap in Figure 1b shows the number of accidents occurring on each day of the week. It reveals a notable trend, with accidents being more frequent on Fridays, indicating a possible correlation between the day of the week and the number of accidents[10].

Modelling and Analysis

In this analysis, the target variable is the severity of traffic collisions, with the goal of predicting whether an accident is property damage (0), injury (1), or severe (2). To achieve this, we compare three machine learning models: Decision Tree, Random Forest, and Gradient Boosting. Each model uses the same set of predictors (weather, road surface conditions, etc). By analysing the performance of these models, we aim to identify the most effective technique for accurately predicting accident severity and informing traffic safety initiatives.

For our analysis, we split the dataset using the standard train-test split approach, allocating 90% of the data for training and reserving the

remaining 10% for testing. This strategy proved effective given the large size of the dataset, ensuring the test set remained representative while providing sufficient data for training the models.

For model training, optimization, and evaluation, we used a combination of libraries. Scikit-learn was the primary library for building and evaluating machine learning models, leveraging its comprehensive tools for classification and performance metrics. Skopt facilitated the Bayesian Optimization process, optimizing hyperparameters efficiently. For visualizing data and results, Matplotlib and Seaborn were utilized to create informative plots and graphs.

For the Decision Tree model, we used the CART framework, a non-parametric method that splits data iteratively to maximize information gain. Its simplicity and interpretability highlight key features driving accident severity predictions. GridSearchCV (5-fold) was used to optimize hyperparameters like max_depth, min_samples_leaf, and min_samples_split. Best Hyperparameters Identified (grid values shown in square brackets):

max_depth [5, 10, 20]	10
min_samples_leaf [1, 5, 10]	10
min_samples_split [2, 5, 10]	2

For the Random Forest model, we applied Bayesian Optimization with BayesSearchCV for efficient hyperparameter selection. Unlike Grid Search, Bayesian Optimization focuses on promising areas in the parameter space, saving computational resources while achieving high accuracy. A 3-fold cross-validation ensured the model's generalizability to unseen data, with accuracy as the primary metric. Best Hyperparameters Identified (ranges shown in brackets):

n_estimators (50, 500)	482
max_depth (10, 20)	12
max_features (5, 10)	10

Additional parameters included class_weight = 'balanced' to handle class imbalance and min_samples_split = 1000 to prevent overfitting.

Gradient Boosting was our third model, chosen for its ability to handle complex data relationships through iterative improvements in prediction accuracy. We used Random Search to identify the best hyperparameters. This method allowed us to efficiently explore the parameter space without assumptions about the underlying model, complementing the systematic approaches used for the other models. Best Hyperparameters Identified (grid values shown in square brackets):

max_depth [2, 5, 10]	5
learning_rate [0.01, 0.05, 0.1]	0.05
n_estimators [10, 50, 100]	100

To evaluate model performance, we utilized a variety of metrics, including Accuracy, Precision, Recall, and F1-score. These metrics provided a comprehensive assessment of each model's performance, balancing overall correctness with the ability to identify classes effectively.

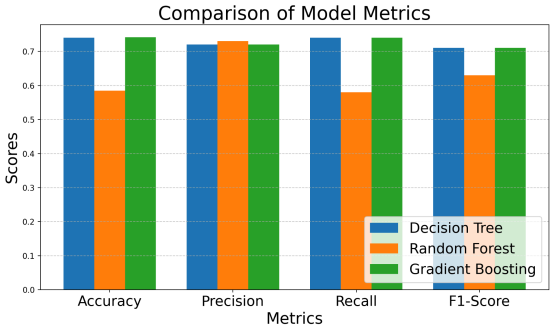

Figure 2. This plot illustrates the Accuracy, precision, recall and F1-score between each model

As shown in Figure 2, among the three models, Gradient Boosting emerged as the most robust, achieving the highest accuracy of 74.13% and a balanced recall of 74%. It matched the Decision Tree in weighted F1-score (0.71) but demonstrated greater consistency, with cross-validation scores averaging 73.95%. The Decision Tree also performed well, with a comparable accuracy of 74.03% and strong recall for property damage cases (93%). However, it struggled

significantly with injury cases (recall: 36%) and performed poorly on severe cases, highlighting limitations in handling minority classes. In contrast, the Random Forest model lagged behind, achieving the lowest accuracy at 58.46%. While it excelled in precision (73%), indicating reliability in predictions, it suffered from low recall (58%), particularly for injury and severe cases. Its weighted F1-score of 0.63 reflects this trade-off, making it the least effective model overall. The Gradient Boosting model's balance of precision, recall, and consistency solidified its position as the best performer. Below are the confusion matrices for each model:

Decision Tree			
	Property Damage	Injury	Severe
Property Damage	11832	688	0
Injury	3746	1865	2
Severe	138	193	1

Random Forest			
	Property Damage	Injury	Severe
Property Damage	7457	4082	981
Injury	927	3132	1554
Severe	31	91	210

Gradient Boosting			
	Property Damage	Injury	Severe
Property Damage	11748	772	0
Injury	3678	1935	0
Severe	131	198	3

The Decision Tree excelled in predicting property damage but struggled with minority classes. The Random Forest model distributed errors more evenly but performed poorly in classifying property damage cases, misclassifying many as injuries or severe cases. However, Random Forest guessed severe more often correctly than others indicating that it was less prone to overfitting. The Gradient Boosting model showed similar strengths to the Decision Tree in class 0 and class 1 predictions, with fewer misclassifications overall, but it struggled slightly with severe cases, misclassifying most as injuries.

In conclusion, we tested various models, but none outperformed the Gradient Boosting model. While the Decision Tree and Random Forest models excelled in certain areas like recall and precision, respectively, Gradient Boosting provided the best overall performance with balanced recall, consistent accuracy, and reliability. Therefore, it was selected as the final model, offering a more robust solution despite its simplicity compared to combined approaches.

Future Works

One direction for future work is incorporating additional features, such as traffic volume, nearby construction zones, or historical accident data, to provide a broader context for accident severity[6, 7, 8]. Moreover, deep learning models, particularly neural networks, could be explored to capture more complex patterns that traditional models may overlook. Finally, addressing imbalanced classes through methods like SMOTE or class-weight adjustments could improve predictions for less frequent classes, such as severe and fatal accidents.

Conclusion

In conclusion, we tested multiple models including Decision Tree, Random Forest, and Gradient Boosting, with Gradient Boosting emerging as the best performer, achieving an accuracy of 74.13%. While this model outperformed the others, there is still room for improvement. Incorporating additional features, exploring deep learning models like neural networks, and addressing class imbalances could improve predictions for minority classes.

References

Methods

1. Elyassami, S., Hamid, Y., & Habuza, T. (2021, June). Road crashes analysis and prediction using gradient boosted and random forest trees. In *2020 6th IEEE Congress on Information Science and Technology (CiSt)* (pp. 520-525). IEEE.
2. Silva, C., & Saraee, M. (2019, November). Predicting road traffic accident severity using decision trees and time-series calendar heatmaps. In *2019 IEEE Conference on Sustainable Utilization and Development in Engineering and Technologies (CSUDET)* (pp. 99-104). IEEE.
3. Chen, M. M., & Chen, M. C. (2020). Modeling road accident severity with comparisons of logistic regression, decision tree and random forest. *Information*, *11*(5), 270.
4. Yan, M., & Shen, Y. (2022). Traffic accident severity prediction based on random forest. *Sustainability*, *14*(3), 1729.
5. Priscilla, C. V., & Prabha, D. P. (2021). A two-phase feature selection technique using mutual information and XGB-RFE for credit card fraud detection. *Int. J. Adv. Technol. Eng. Explor*, *8*(85).

Applied Problem

6. Li, Y., Sun, X., Wang, Y., & Zhang, W. (2023). Predicting crash severity on mountain freeways using dynamic traffic and weather data. *Transportation Safety and Environment, 5*(4), tdad001.
7. Chen, H., Lu, W., & Wu, H. (2020). Investigating the impacts of road geometry and traffic flow on accident severity. *Journal of Transportation Engineering, 146*(12), 04020118.
8. Abadi, A., & Nix, S. (2023). Nonlinear effects of traffic statuses and road geometries on highway traffic accident severity. *PLOS ONE, 18*(3), e0314133.
9. Nagy, G., & Kiss, B. (2022). The impact of road geometric formation on traffic crashes and their severity levels: A Budapest case study. *Sustainability, 14*(14), 8475.
10. Kumar, S., & Tiwari, R. (2022). Influence of road infrastructure design over traffic accidents: A simulated case study. *Infrastructures, 9*(9), 154.
11. Borucka, A., Kozłowski, E., Oleszczuk, P., & Świderski, A. (2020). Predictive analysis of the impact of the time of day on road accidents in Poland. Open Engineering, 11(1), 90–99.
12. Yilmaz, A., & Yilmaz, S. (2020). Variations in traffic accidents on seasonal, monthly, daily, and hourly basis: Eskisehir case. *International Journal of Environmental Research and Public Health, 17*(12), 4398.
13. Wang, L., Zhang, X., & Liu, Y. (2022). Prediction of seasonal variation in traffic collisions on urban highways. In *2022 IEEE International Conference on Intelligent Transportation Systems (ITSC)* (pp. 867–874). IEEE.

Data

14. https://www.kaggle.com/datasets/jonleon/seattle-sdot-collisions-data
15. https://www.seattle.gov/Documents/Departments/SDOT/GIS/Collisions_OD.pdf

Python Packages

16. https://scikit-learn.org/1.5/api/sklearn.feature_selection.html#module-sklearn.feature_selection
17. https://xgboost.readthedocs.io/en/stable/get_started.html
18. https://scikit-learn.org/1.5/api/sklearn.model_selection.html
19. https://scikit-learn.org/1.5/api/sklearn.ensemble.html
20. https://scikit-learn.org/1.5/modules/model_evaluation.html
21. https://scikit-optimize.github.io/stable/auto_examples/sklearn-gridsearchcv-replacement.html

Appendix

Student	Contribution to Project
Nathen Hale Fernandes	Project Proposal, Model comparison Video Part of brain storming,Research for Gradient Boosting for report, Gradient Boosting Code, Model Comparison part of final Video, Model Comparison of final report
Alexander Armstrong	Project Proposal, Feature Selection Video Part, Research for Decision Tree for first report, Decision Tree Code, Final Video Model Planning Explanations, Various Graphs for Report, Model Explanation for Final Report
Marc Crasto	Project Proposal, Introduction and Description of Available Data for Brainstorming Video, Research for Random Forest Model for Report Draft, Code for Random Forest Model, Introduction and Data Preprocessing for Group Presentation Video, Data Preprocessing Section for Final Report.