Analyze United States Road Accident And Classify using Machine Learning Algorithms and Resampling Techniques

Nafea Ebrahem Yazan Alzaibak mohamed altanoukhi

Abstract

Accident severity classifications play a crucial role in improving traffic safety and enhancing the efficiency of emergency response systems. In this study, machine learning techniques were used to classify accident severity based on.

A representative sample (including 500,000 records) from the U.S. accident database for the years 2016–2023 was used. One of the key preprocessing steps to improve model performance was converting the original severity variable into a binary classification task, where the severity levels 1 and 2 were combined into "non-severe accident," and levels 3 and 4 were combined into "severe accident.

The dataset contained multiple features such as weather, vision, temperature, time of the accident, and type of road. After applying preprocessing steps, multiple classification models were tested, including Random Undersampling and synthetic data generation methods such as ADASYN and SMOTE. Several models were evaluated, including Random Forest and Linear Regression, and results were compared using CatBoost and XGBoost models.

The results showed that ensemble methods (like CatBoost and XGBoost) achieved the highest accuracy, while synthetic data generation methods improved balance between classes. These results confirm the importance of using artificial intelligence techniques in predicting accident severity, helping decision-makers in the transport sector.

I. INTRODUCTION

Traffic accidents are a major source of concern for public safety worldwide, as they cause thousands of deaths and injuries annually. Given their high level of danger, it is of utmost importance to develop adaptive safety response systems. With the increasing availability of traffic data, it has become possible to apply advanced machine learning models, which are among the most powerful tools for analyzing complex patterns and predicting accident outcomes.

We conducted an analytical process for our data since the project is divided into analysis and classification tasks. We began by describing the dataset and examining each column of information, identifying its type and known domain. We also created numerical plots of our data and described the relationships between numerical columns. Furthermore, we plotted correlations between variables and focused specifically on the "Severity" column, which represents the severity of the accident. We identified its values, their meanings, and the number of entries for each category in order to understand the data more effectively.

From the initial results, it became clear that there was a significant imbalance between the severity categories, as most cases belonged to the non-severe accident classes (such as minor property damage or minor injuries), while the severe accident cases represented only a small fraction. This imbalance directly affects the performance of machine learning models, as they tend to be biased toward predicting the majority class and thus perform poorly when predicting rare but critical cases.

To address this challenge, we applied resampling techniques such as oversampling and undersampling to balance the classes and ensure that the models could learn patterns more effectively. This step is essential to improve model accuracy, particularly when predicting severe accidents that are associated with rare but influential factors like adverse weather conditions or time-related attributes.

In this paper, we present a robust classification framework to predict accident severity using ensemble learning algorithms (CatBoost, Random Forest, Logistic Regression, LightGBM, XGBoost).

We also applied resampling techniques (SMOTE, NearMiss, RandomUnderSampler, ADASYN).

Since oversampling techniques may lead to overfitting on minority classes, we implemented them carefully while also applying model optimization methods, such as hyperparameter tuning, to enhance model performance and reduce bias.

Our approach involved converting the original multi-class severity variable into a binary classification problem (class 1,2 → not severe; class 3,4 → severe).

The aim of the project is to analyze traffic accident data, improve model performance, and reduce class imbalance. This contributes to building an effective early warning system capable of predicting severe accidents and supporting decision-making in the transportation sector, particularly by identifying rare but critical cases that are often influenced by adverse weather or time-related attributes.

II. Related Work

Classifying the severity of traffic accidents is among the most important applications of artificial intelligence in the field of road safety. Numerous studies have focused on developing predictive models to determine accident severity, based on multiple environmental and situational variables such as weather, visibility, road conditions, and time of occurrence.

Sobhan Moosavi and his team proposed a model known as DAP (Deep Accident Prediction). The model achieved high F1-score values through the use of deep learning applications, despite the challenges posed by imbalanced datasets. The authors emphasized the importance of advanced preprocessing methods, such as data cleaning and preparation, using the U.S. accident database, which contains a very large number of records.

Almalki et al. (2019) developed a model for classifying traffic accident severity using machine learning techniques. Their dataset included accident characteristics and traffic rules. The researchers tested multiple algorithms, including Naive Bayes, Random Forest, and SVM. They aimed to predict the severity of traffic accidents with higher accuracy, and their analysis concluded that selecting the most influential features plays a major role in determining severity levels, leading to improved model performance.

Sharma et al. (2023) conducted a study that relied on two models: LightGBM and Balanced Bagging Classifier.

The aim of predicting traffic accident severity has led researchers to use road accident databases and apply class balancing techniques to address class imbalance.

To ensure effective model training, the results showed that the ensemble model outperformed individual models in terms of accuracy and F1-score, proving its effectiveness in dealing with imbalanced data problems.

III. Methodology

.Dataset Overview

Data Source:
A dataset of road accidents in the United States (US Accidents 2016–2023) was used. We selected a sample of 500,000 records, which represents nearly half a million accident cases collected from multiple sources in the U.S., such as government agencies, road sensors, and user reports.

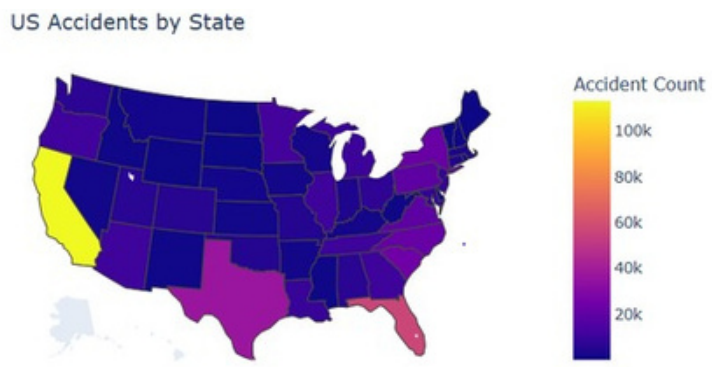
Dataset Structure:
Number of records: 500,000
Features:
The dataset includes numerical and categorical features, such as:
Numerical features: accident severity, distance, speed, visibility, humidity, temperature, wind speed, air pressure, duration of rainfall, and snowfall.
Categorical features: accident location, wind direction, sunrise/sunset times.

Target Variable:
The original severity variable had four levels (1 to 4). It was transformed into two classes:
Not Severe (1 & 2): Low severity.
Severe (3 & 4): High severity.

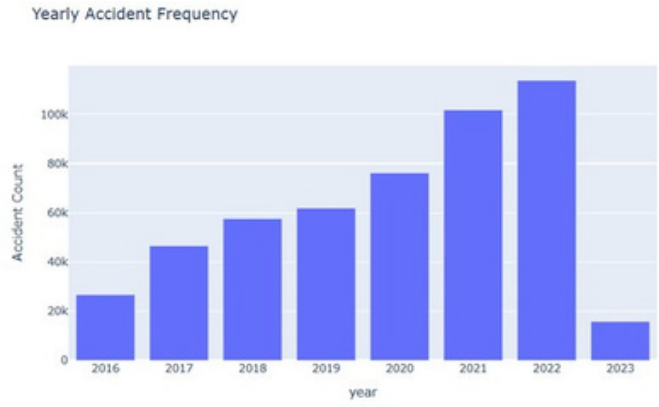
..Data mining & Preprocessing

Descriptive Data Analysis:
This section presents a detailed analysis of the dataset structure: duplicate values, missing values, and statistical distributions. This helped guide the preprocessing stages.
We displayed our dataset and identified the data type for each column, along with the number of missing values in each attribute.
We also plotted histograms to verify the distribution of numerical attributes, many of which contained numerous missing values.
For each numerical attribute, we used the **describe()** function to extract summary statistics, including:
mean, median, standard deviation, minimum, maximum, and outliers.
We also plotted boxplots to visualize numerical attribute distributions and detect outliers.

we presented a plot showing the distribution of accidents across 49 states within the United States.



We also have this chart that shows the distribution of accidents within each year :



Handling Missing Values:
Missing values in numerical features were filled using the median.
For categorical features, the most frequent value was used.
Feature Engineering:
Converted Start_Time and End_Time into date-time format.
Extracted new attributes such as Duration_Minutes.
Extracted additional features like month, day of the week, and hour from the date-time.
Converted the Is_Weekend column into a Boolean feature (1 for weekends, 0 for weekdays).
Encoded categorical variables such as Weather_Condition into simplified categories (e.g., clear, cloudy, rain, snow, fog, storm).
Simplified wind directions in the Wind_Direction column by grouping them into:
WEST → W, EAST → E, SOUTH → S, NORTH → N.
Replaced rare categories in categorical variables with "Other".
Categorical missing values were filled with "Unknown".
Correlation Analysis:
Applied Chi-Square (χ^2) test for correlations between categorical variables.
Used correlation matrix to analyze relationships between numerical features.
Handling Outliers:
Outliers were treated using the Interquartile Range (IQR) method.
Outlier boundaries were calculated as follows:
Upper Bound = Q3 + 1.5 × IQR
Lower Bound = Q1 – 1.5 × IQR
For continuous numerical variables such as:
Distance(mi), Temperature(F), Wind_Chill(F), Pressure(in), Wind_Speed(mph),
values were clipped using np.clip().
For variables like Visibility(mi) and precipitation (Precipitation(in)), extreme values were adjusted based on acceptable ranges (e.g., precipitation clipped at 0.99).
Summary:
These preprocessing steps — handling missing values, feature engineering, correlation analysis, and outlier treatment — contributed to improving data quality, reducing noise, and enhancing model learning ability, which ultimately supports more accurate severity prediction.

The StandardScaler was applied to the numerical features as part of the processing pipeline. Additional scaling techniques such as QuantileTransformer were also used, where features with skewed distributions were transformed using output_distribution='normal'. This was applied to the columns Precipitation(in), Visibility(mi), and Pressure(in).
The PowerTransformer (Yeo-Johnson method) was used to transform features with negative or zeroes values to a distribution closer to normal, and it was applied to Distance(mi) and Duration_Minutes.
A square root transformation (np.sqrt) was applied to Wind_Speed(mph) to reduce the impact of large values and help approximate a normal distribution.

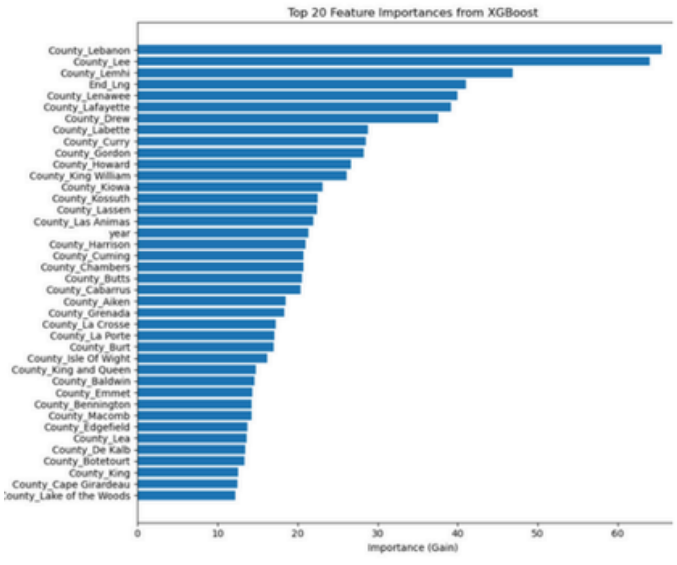
Model Selection and Training including Resample techniques:

In the pipeline, we applied StandardScaler on the numerical data, and we applied OneHotEncoding on the categorical data and we make sure that the dataset didn't have a large unique values columns.
This was applied using MostFrequentUnderSampler, and this was applied on the dataset.
After that, we applied it on models such as (RandomForest, XGBoost, Logistic, LightGBM, CatBoost), and we tried RandomForest with GridSearch in order to choose the best hyperparameters.
As for oversampling, we tried several techniques, but all of them did not benefit in the problem because we applied SMOTE and ADASYN, but they did not help since the minority classes remained and didn't trained well. Even with several attempts to adjust the hyperparameters, we did not find a result where the macro F1 exceeded 0.61.
Therefore, we chose a balanced method, which is converting the multi-class classification (1 → 4) into binary classification (severe, not severe).
Specifically, we merged 1 and 2 into not severe, 3 and 4 into severe.
As we mentioned before, we applied UnderSampler, but after applying it on severe and not severe, it did not give good results because NearMiss and other techniques did not help.
As for hyperparameters, we also tried different values, but with UnderSampler random, we found better results.
The best hyperparameters for XGBoost were:
n_estimators=10000, max_depth=8, learning_rate=0.01, subsample=0.8, colsample_bytree=0.8, min_child_weight=1.
And the results before applying undersampling were bad for recall and good for precision, but we care about recall to be higher and more accurate, because FN cases are costly in our case, since classifying an accident as not severe, despite being severe, is very critical.

Feature Selection:

When we used XGBoost, we applied model.feature_importance to get the top 20 features.
After that, we applied SelectPercentile in order to keep the most important features, but the model did not show any difference as changing its values compared to the remaining features.
The values we tested were (50, 70, 30).
It should be noted that we also performed a manual feature selection process, where we removed the columns that had no importance in the data.

TOP 20 FEATURES



Evaluation Metrics

results after applying OVERSAMPLING on catBOOST with multiclassification

	precision	recall	f1-score	support
1	0.49	0.72	0.58	855
2	0.89	0.90	0.90	79628
3	0.59	0.56	0.58	16904
4	0.36	0.30	0.32	2613
accuracy			0.83	100000
macro avg	0.58	0.62	0.60	100000
weighted avg	0.83	0.83	0.83	100000

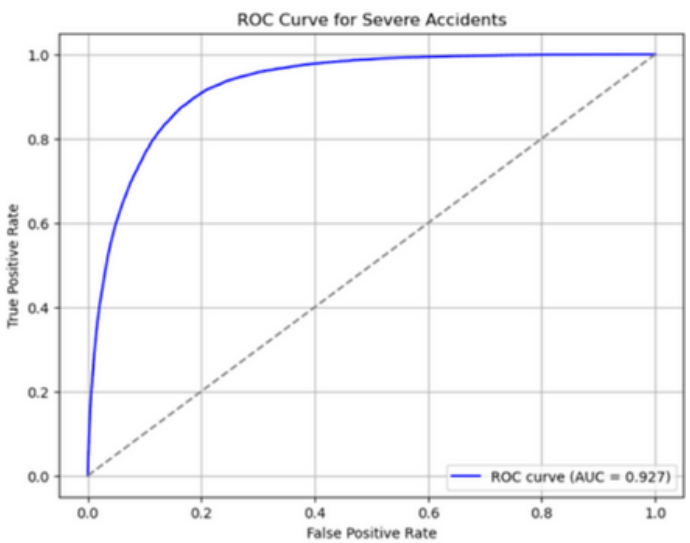
results before applying UNDERSAMPLING on XGBoost with Binary classification (severe, not severe)

	precision	recall	f1-score	support
Not_Severe	0.91	0.95	0.93	80483
Severe	0.75	0.61	0.67	19517
accuracy			0.88	100000
macro avg	0.83	0.78	0.80	100000
weighted avg	0.88	0.88	0.88	100000

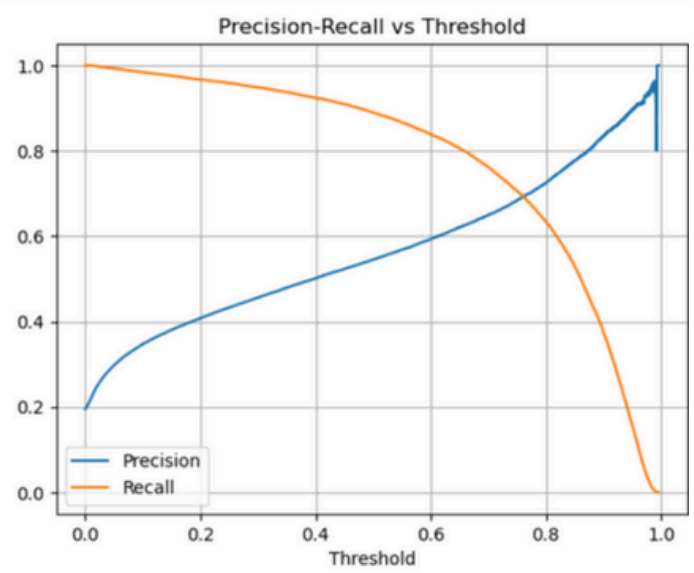
results after applying UNDERSAMPLING on XGBoost with Binary classification (severe, not severe)

	precision	recall	f1-score	support
0	0.97	0.82	0.89	80483
1	0.54	0.89	0.68	19517
accuracy			0.83	100000
macro avg	0.76	0.85	0.78	100000
eighted avg	0.89	0.83	0.85	100000

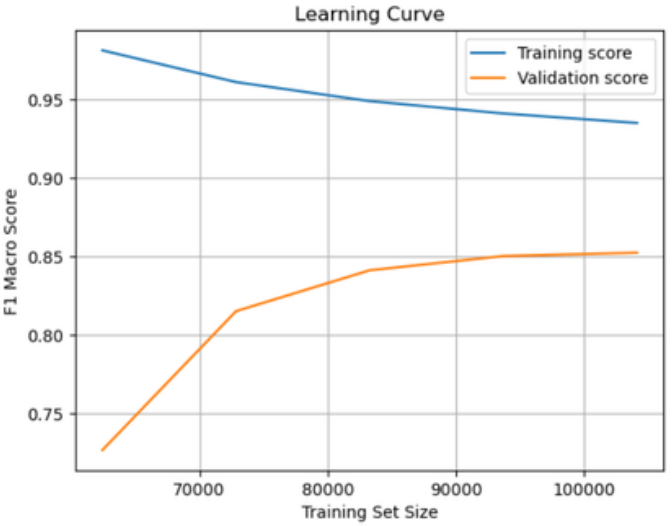
AUC,ROC



precision - recall vs (threshold)



Learning Curve



IV. Conclusion and Future Work

The aim of this study was to classify traffic accidents into severe and non-severe using machine learning models and an imbalanced dataset from the real world.
Comparison of the solution with baseline or SOTA studies:

1. Simple Literature Review:
The problem of traffic accidents has been addressed in detail in recent years for several reasons, including the danger, causes, and recurrence, in addition to the lack of solutions. We found several studies or papers:
1- Traffic crash severity using hybrid of balanced bagging classification and light gradient boosting machine
A Hybrid of Balanced Bagging Classifier + LightGBM was used:
Accuracy: 77.7%, Precision: 75%, Recall: 73%, F1-Score: 68%

2- A classification and recognition model for the severity of road traffic accident
SVM (Support Vector Machine) + Rough set were used.
Rough set increased accuracy.
In general, the works focused on improving the main metrics such as accuracy, F1-score, and recall, using different machine learning techniques, whether shallow or deep, with varying attention to data quality and feature generalization.

2. Comparison of the results with published studies:

Study/Method	Accuracy	F1-score	Notes
Paper [1]	77.7 %	0.68	Based on BBC + LGBM model
Paper [2]	88.7 %	-	Based on Rough set + SVM model
Proposed Solution	89.0 %	0.79	Using XGBClassifier with certain adjustments

It is clear from the table that the proposed solution surpasses the baseline approaches in terms of accuracy and F1-score, especially in cases where the class balance is skewed in the inputs.

3. Position of the current work compared to the baseline studies:

Exceeding previous studies:
In terms of overall performance, such as accuracy and evaluation metrics, the proposed solution demonstrates improvement because it incorporates techniques used in previous studies, introduces additional engineered features, and applies sampling methods to address the imbalance class problem. Furthermore, the merging of target class categories also contributed to enhancing the dataset, and altogether, these factors improved the overall performance.

Strengths:
The proposed design performs effectively when tested on new datasets, showing strong generalization capabilities. It also takes into account important factors such as execution speed and result interpretation, which add practical value to its implementation. In addition, the solution addresses common challenges without requiring a complete retraining process, and it is able to operate efficiently even on very large datasets, which highlights its scalability.

Areas that can be improved:
Despite these strengths, there remain areas for further enhancement. Reducing false negatives (FN) should be prioritized, as these errors carry significant consequences in classification tasks. Further experiments should be conducted on more diverse datasets to ensure broader applicability and robustness. Finally, exploring other deep learning approaches could provide additional avenues for improvement and potentially raise performance to even higher levels.

Among the tested models, XGBoost achieved the best performance with an F1-score = 79, and the sampling techniques such as RandomUnderSampler improved recall, making the model more reliable in detecting severe accidents, where the severe recall reached 89.

Future Work:
We can integrate more real-time features, such as traffic volume or camera data.

References:

Sampled Dataset:

[1] J. T. Tonny, "US Accidents – March 2023," Kaggle, Dataset, Mar. 2023. [Online]. Available: https://www.kaggle.com/datasets/joytuntonny/us-accidents-march23

Main Dataset:

[2] S. Moosavi, M. H. Samavatian, S. Parthasarathy, and R. Ramnath, "A Countrywide Traffic Accident Dataset," Kaggle, Dataset, (data from Feb. 2016–Mar. 2023). [Online]. Available: https://www.kaggle.com/datasets/sobhanmoosavi/us-accidents

Paper 1:

[3] X. Jianfeng, G. Hongyu, and T. Jian, "A classification and recognition model for the severity of road traffic accident," Advances in Mechanical Engineering, vol. 11, no. 5, pp. 1–8, May 2019. doi: 10.1177/1687814019851893

Paper 2:

[4] (Authors not specified), "Predicting traffic crash severity using hybrid of balanced bagging classification and light gradient boosting machine," ResearchGate, 2023. [Online]. Available: https://www.researchgate.net/publication/367664188_Predicting_traffic_crash_severity_using_hybrid_of_balanced_bagging_classification_and_light_gradient_boosting_machine

[5] T. Chen and C. Guestrin, "XGBoost documentation," XGBoost, [Online]. Available: https://xgboost.readthedocs.io/en/stable/index.html

[6] F. Pedregosa et al., “Scikit-learn: Machine Learning in Python,” Journal of Machine Learning Research, vol. 12, pp. 2825–2830, 2011. [Online]. Available: https://jmlr.org/papers/volume12/pedregosa11a/pedregosa11a.pdf

[7] M. Mouaici and F. Royet, "Weighted Classification Model to Predict Traffic Accident Severity," inProc. 2023 9th Int. Conf. Control, Decision and Information Technologies (CoDIT), Jul. 2023. doi: 10.1109/CoDIT58514.2023.10284295