Yuchen Tao

SML_report (2).pdf

21 任选一个空白的提交。每个入口可查3次 查完请用其他文件覆盖掉 注意保护个人信息。如果不出结果 换空白的入口提交（打开里面没有提交...

bb

University of Lagos

Document Details

Submission ID
trn:oid:::1:3376407124

Submission Date
Oct 17, 2025, 11:17 AM GMT+1

Download Date
Oct 17, 2025, 11:18 AM GMT+1

File Name
SML_report_2_.pdf

File Size
989.3 KB

4 Pages

2,344 Words

13,026 Characters

*% detected as AI

AI detection includes the possibility of false positives. Although some text in this submission is likely AI generated, scores below the 20% threshold are not surfaced because they have a higher likelihood of false positives.

Caution: Review required.

It is essential to understand the limitations of AI detection before making decisions about a student's work. We encourage you to learn more about Turnitin's AI detection capabilities before using the tool.

Frequently Asked Questions

How should I interpret Turnitin's AI writing percentage and false positives?
The percentage shown in the AI writing report is the amount of qualifying text within the submission that Turnitin's AI writing detection model determines was either likely AI-generated text from a large-language model or likely AI-generated text that was likely revised using an AI paraphrase tool or word spinner.

False positives (incorrectly flagging human-written text as AI-generated) are a possibility in AI models.

AI detection scores under 20%, which we do not surface in new reports, have a higher likelihood of false positives. To reduce the likelihood of misinterpretation, no score or highlights are attributed and are indicated with an asterisk in the report (*%).

The AI writing percentage should not be the sole basis to determine whether misconduct has occurred. The reviewer/instructor should use the percentage as a means to start a formative conversation with their student and/or use it to examine the submitted assignment in accordance with their school's policies.

What does 'qualifying text' mean?
Our model only processes qualifying text in the form of long-form writing. Long-form writing means individual sentences contained in paragraphs that make up a longer piece of written work, such as an essay, a dissertation, or an article, etc. Qualifying text that has been determined to be likely AI-generated will be highlighted in cyan in the submission, and likely AI-generated and then likely AI-paraphrased will be highlighted purple.

Non-qualifying text, such as bullet points, annotated bibliographies, etc., will not be processed and can create disparity between the submission highlights and the percentage shown.

Modelling Regional Accident Risk with feature Learning

Ziqi Wen (1129440) Yiru Liu (1350975) Jing Wang (1202659)
The University of Melbourne — COMP90051 Statistical Machine Learning

1 Introduction

Road accidents still remains as a major concern for public safety worldwide nowadays. To make roads safer and possibly help governments plan cities more effectively, it's essential to understand what causes accidents and how severe they are. In this project, we will employ machine learning models to model and forecast critical accident regions by performing data aggregation and integrating spatial, environmental, and infrastructural attributes derived from past accidents in 2023, sourced from the US-Accidents dataset (https://www.kaggle.com/sobhanmoosavi/us-accidents).

Our research question is: Is it possible to model and forecast long-term accident hotspots by integrating spatial, environmental, and infrastructural characteristics from the US-Accidents dataset, and which factors most significantly influence regional accident risk?

2 Literature Review

US-Accidents Dataset (Moosavi et al., 2019)

Moosavi et al. (2019) [1] presented the US-Accidents dataset, a comprehensive national resource aimed at mitigating the deficiency of extensive, publicly accessible, and contextually rich accident data at that time. From 2016 to 2019, the authors got real-time traffic reports from the MapQuest and Big Traffic APIs. They reverse the geocoding by using the location data from Nominatim, the weather data from the Weather Underground API, and the points of interest (infrastructure features) from OpenStreetMap that are within an optimized spatial threshold. The final dataset, which will be updated until 2023, includes over 2.2 million accidents and 45 original features. This makes it possible to do large-scale spatiotemporal modeling and a wide range of research.

Deep Accident Prediction Model (Moosavi et al., 2019)

Moosavi et al. (2019) [2] expanded their initial research and introduced the Deep Accident Prediction (DAP) model, a deep learning framework designed for precise, real-time accident risk forecasting. They used the US Accidents Dataset to build and train DAP with recurrent LSTM, fully connected layers, and trainable spatial embeddings all at once. They then test the model in six US cities, and DAP beats the traditional machine learning baseline. This paper shows how we can use and change spatial, environmental, and infrastructural information together. This gives us a methodological basis for using FT-transformer in this project.

Revisiting Deep Learning Models for Tabular Data (Gorishniy et al., 2021)

Gorishniy et al. (2021) [3] presented the FT-Transformer, which is a modification of the Transformer architecture designed for tabular data. Inside the model, a feature tokeniser module converts each numerical and categorical feature into a learnable embedding. The resulting tokens, along with a special [CLS] token, are then processed by self-attention layers to make inference. The model performed at the cutting edge, or better than, gradient-boosted decision trees and other deep models across a wide range of tabular datasets. However, the paper primarily focuses on firm performance and simplicity rather than directly estimating feature importance.

3 Method

Preprocessing and Feature Engineering

We first filtered the US Accidents dataset to only include records from 2023 to make sure the time was consistent. Then we got rid of the highly correlated coordinate features (End_Lat, End_Lng) and the instances with more than 50% missing values. For categorical features, N/A is used to fill in the remaining missing values, and for numerical features, the median is used.

We built features in four different ways:

1. Time: We get the hour, day, month, and weekday from Start_Time and add cyclic encoding (sin, cos). We make the features rush hour, weekend flags, seasonal indicators, accident duration, and weather timestamp lag manually.
2. Spatial/Infrastural: We one-hot encode the top 30 cities and counties, binary-encode the street and zip code, and then binarize all the boolean infrastural features (Junction, Bump, Traffic_Signal, etc.).

3. Environmental: We group more than 70 different weather conditions into six main classes (Clear, Cloudy, Rain, Storm, Snow, and Fog). We also use cyclic encoding on Wind Direction, log-transform skewed variables (Visibility, Wind Speed), make a precipitation flag, and binarize twilight indicators.
4. Textual: We use Sentence-Transformer (all-MiniLM-L6_v2) to make 384-dimensional sentence embeddings of Description. Then, we use PCA to cut them down to 50 main components.

Finally, all features were aggregated by spatial $0.1° \times 0.1°$ grid using geocoordinate feature, compute class labels (Severity_count, Severity_mean, Severity_max) of different sematical meanings with log-transforms applied to mitigate skewness.

Machine Learning Models

We compared three regression models with increasing representational complexity:

Ridge Regression (Linear baseline): a regularized linear model minimizing squared error with L2 penalty, chosen for its interpretability and ability to capture global linear trends between spatial–environmental features and accident intensity.

XGBoost Regressor (Gradient-Boosted Trees): an ensemble of decision trees trained sequentially to minimize residual errors via gradient boosting with learning rate and depth control. XGBoost effectively models nonlinear dependencies and heterogeneous feature interactions (e.g., between weather, time, and infrastructure).

FT-Transformer (Tabular Transformer): The FT-Transformer model uses **multi-task regression** to predict the number of accidents, the average severity, and the worst severity at the same time. The first step is to turn each "feature block" (like weather, time, POI, city) into a "token representation." Continuous features are linearly projected to a shared embedding dimension, and categorical groups are mapped through learned embeddings. A learnable **[CLS] token** is added to the beginning of the input sequence to get the global context. The input sequence is ([CLS, weather, time, POI, city, county, wcat]). A stack of Transformer encoder layers uses multi-head self-attention and feed-forward networks to model how different features interact with each other. A regression head (LayerNorm $\rightarrow$ Linear $\rightarrow$ GELU $\rightarrow$ Dropout $\rightarrow$ Linear) takes the final [CLS] embedding and gives three normalized target values.

Nested Cross Validation Setup

We used a custom nested cross-validation pipeline that is shared by all three models Ridge Regression, XGBoost, and FT-Transformer to ensure fairly comparison and without bias.

For the Nested Cross Validation setup, the outer loop (10-fold) estimated how well the model would work on new data, and the inner loop (3-fold) chose the best hyperparameters. The configuration with the lowest mean absolute error (MAE) was then retrained on the complete outer-training data and tested on the held-out fold.

Tuned Hyperparameters: For **Ridge Regression**, we searched over regularization strengths ($\alpha \in [10^{-3}, 2\times10^{2}]$). For **XGBoost**, tuning focused on tree complexity and learning rate, with (max_depth $\in \{3, 4, 5\}$) and (learning_rate $\in \{0.2, 0.25, 0.3\}$). For the **FT-Transformer**, we optimized the token embedding dimension, number of encoder layers, and learning rate across (d_token $\in \{64, 128, 192\}$), (n_layers $\in \{2, 3, 4\}$), and (lr $\in \{$1e-3, 1.5e-3, 2e-3$\}$).

To keep the data distribution across folds the same in a regression setting, targets were quantile-binned into 10 groups before being split. The statistical distribution of the response variable (e.g., Severity_mean, Severity_count, and Severity_max) was kept the same across folds. Finally, to prevent data leakage, all features were kept the same across all training folds.

Evaluation Metrics

We gave three different regression metrics that work together:

1. Root Mean Squared Error (RMSE) is a measure of overall prediction accuracy that is sensitive to large deviations.
2. Mean Absolute Error (MAE) is the average difference between the predicted and actual risk of an accident.
3. The Coefficient of Determination (R^2) shows how much of the model's variance can be explained by the mean baseline.

The results for each target variable (Severity_count, Severity_mean, Severity_max) were averaged over the 10 outer folds, and the variability in performance was shown as mean $\pm$ standard deviation.

To examine the Ridge Regression and XGBoost models' generalisation more thoroughly, using the best hyperparameter for each, we created manual learning curves by training on increasingly larger portions of the dataset (5% to 100%) and evaluating with 5-fold cross-validation at each stage. These curves show how the training and validation RMSE change as the sample size grows. This helps find underfitting, overfitting, and data sufficiency.

4 Result

Model Performance Analysis

Overall, all three models performed excellently on Severity_count and Severity_mean, with RMSE values below 0.07 and R-squared values generally above 0.5. The results obtained demonstrate that combined accident frequency and mean severity can be accurately inferred employing spatial, environmental, and infrastructural characteristics. XGBoost and FT-Transformer are both marginally better than Ridge Regression, which shows the necessity to use a nonlinear model to capture complex feature interaction.

Model	Target	RMSE ($\pm$ SD)	MAE ($\pm$ SD)	R^2 ($\pm$ SD)
Ridge Regression	Severity_count	0.090 ± 0.002	0.074 ± 0.002	0.286 ± 0.027
	Severity_mean	0.061 ± 0.004	0.035 ± 0.002	0.504 ± 0.038
	Severity_max	0.674 ± 0.019	0.486 ± 0.013	0.368 ± 0.028
XGBoost	Severity_count	$\mathbf{0.036 \pm 0.001}$	$\mathbf{0.027 \pm 0.001}$	$\mathbf{0.887 \pm 0.005}$
	Severity_mean	0.064 ± 0.004	0.033 ± 0.002	0.510 ± 0.041
	Severity_max	0.605 ± 0.019	0.410 ± 0.012	0.485 ± 0.034
FT-Transformer	Severity_count	0.0458 ± 0.0007	0.0353 ± 0.0006	0.8187 ± 0.0051
	Severity_mean	0.0627 ± 0.0008	0.0276 ± 0.0005	0.5242 ± 0.0118
	Severity_max	0.5948 ± 0.0064	0.3483 ± 0.0062	0.5031 ± 0.0088

Table 1: Performance comparison of Ridge Regression, XGBoost, and FT-Transformer across all targets.

On the other hand, all models performed relatively poorly on Severity_max, with R-squared values below 0.5. This means they weren't particularly effective at detecting rare, extreme-severity events. This is likely due to the maximum-severity records exhibiting a heavy-tailed, sparse distribution, which implies that extreme accidents occur rarely and are influenced by factors not included in the dataset (such as driver behaviour or emergency response). Therefore, although the models work well for long-term and average accident patterns, it remains challenging to predict rare, high-impact cases due to data imbalance and missing explanatory variables.

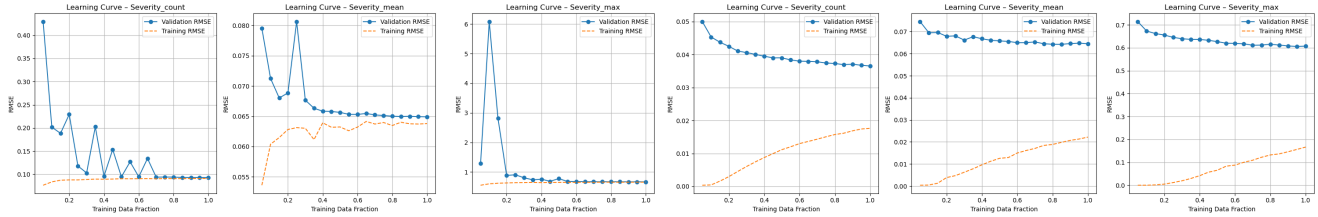

Figure 1: Learning curves for Ridge Regression (left) and XGBoost (right) showing training and validation RMSE across increasing data fractions.

The learning curves in Figure 1 highlight apparent differences in model capacity and generalisation. For Ridge Regression, the RMSE on training and validation remains relatively high for Severity_count and converges as the training size increases. This demonstrates that the model is slightly underfitting, suggesting the linear form can't fully capture the nonlinear patterns in the data, even with more samples. On the other hand, XGBoost has a small but consistent gap between training and validation RMSE, and both values decrease as the training data size increases. The smooth convergence and lack of divergence suggest that the model is generalising well and not overfitting. Validation performance stabilises gradually, indicating that the model's complexity is well-regularised and that it can learn from larger training sets without memorising noise.

Feature Importance Analysis

The feature importance analyses across all models demonstrate the interactions among environmental, temporal, and textual factors in elevating accident risk. Each model employs a distinct approach to achieve this outcome.

The Ridge Regression model assigns greater importance to continuous variables, including Temperature (F), Wind Chill (F), and twilight indicators. This focus is on smooth, global linear relationships. Description embeddings (Emb_PCA_0–6) provide some utility, indicating that textual cues can enhance information retrieval even within a linear framework. This behaviour occurs because linear models primarily exhibit additive effects. Continuous variables with larger numeric ranges yield stronger gradient responses, while boolean or categorical variables, after one-hot encoding, contribute only minor, fixed shifts.

Feature	ΔRMSE (mean over folds)
Emb_PCA_0	0.086090
Emb_PCA_4	0.067645
Emb_PCA_6	0.030021
Emb_PCA_2	0.010963
Emb_PCA_3	0.007446
Emb_PCA_8	0.005987
Emb_PCA_5	0.004327
Emb_PCA_1	0.004063
Emb_PCA_10	0.002963
Emb_PCA_31	0.002634
⋮	⋮

Table 2: Top numeric features ranked by ΔRMSE (mean over folds).

The XGBoost model demonstrates more intricate nonlinear relationships. For instance, is_RushHour_Evening, is_Weekend, Weather_Cloudy, and Traffic_Signal serve as significant predictors. Tree-based methods inherently prefer discrete features, as their threshold-based splits demonstrate the interaction between categorical and continuous conditions.

The FT-Transformer primarily relies on text-embedding features (Emb_PCA_0, Emb_PCA_4, Emb_PCA_6) and, to a lesser degree, on infrastructure or lighting variables (Bump, Astronomical_Twilight_binary), a shown in Table 2. This suggests that the

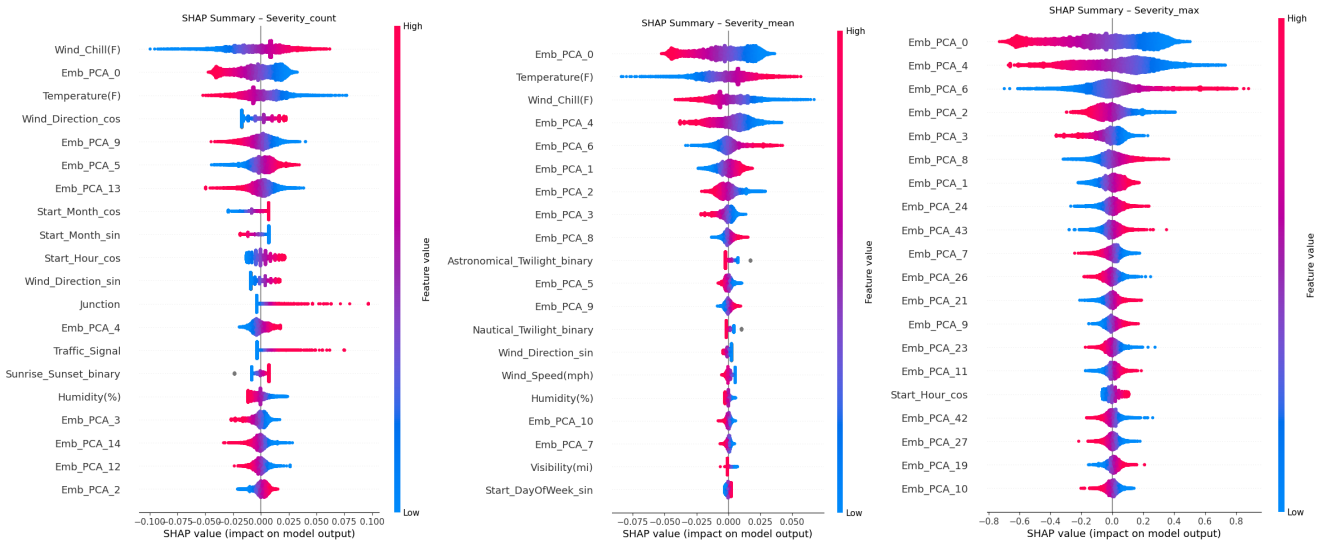

Figure 2: SHAP summary plots for Ridge Regression across Severity_count, Severity_mean, and Severity_max.

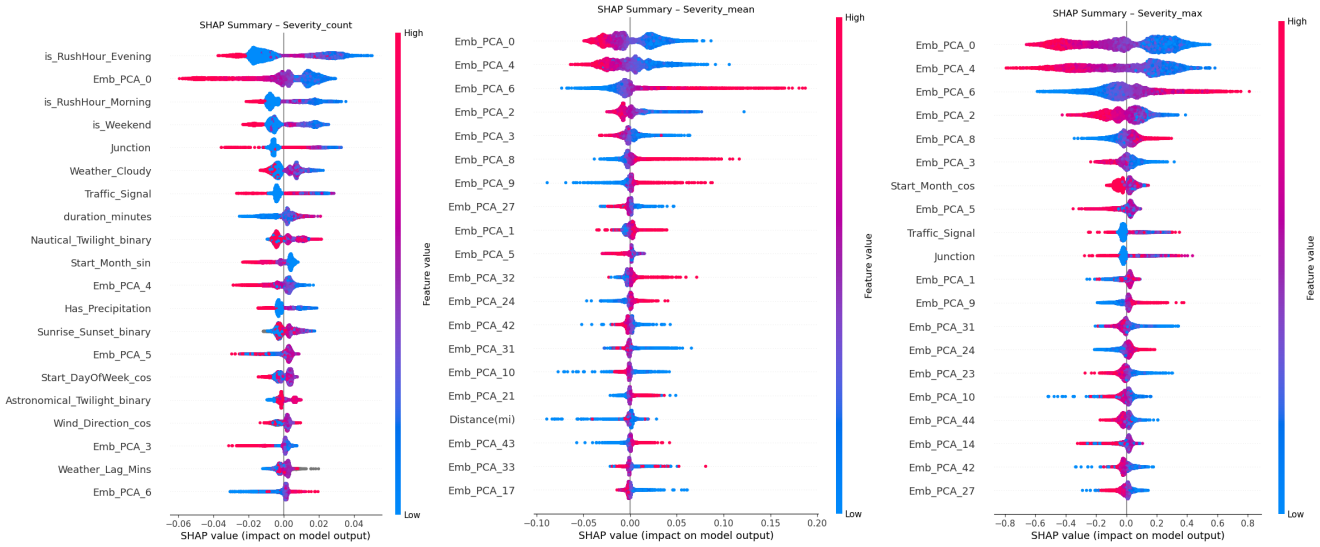

Figure 3: SHAP summary plots for XGBoost across Severity_count, Severity_mean, and Severity_max.

transformer effectively utilises latent semantic representations instead of explicit, handcrafted attributes.

Ridge analyses global numeric trends, XGBoost integrates structured and categorical effects via nonlinear splitting, and FT-Transformer identifies deep contextual patterns. These findings illustrate a spectrum from interpretable linear dependencies to a comprehensive understanding of regional accident risk.

5 Conclusion

This study shows that combining spatial, environmental, and textual features enables accurate modelling of long-term accident hotspots. XGBoost performed best overall, while the FT-Transformer performed similarly and offered better semantic understanding from textual embeddings. These findings demonstrate that integrating structured and contextual information can effectively identify critical factors influencing regional accident risk and facilitate data-driven safety planning.

References

[1] S. M. Moosavi, D. Jung, and A. T. Betke, "US-Accidents: A countrywide traffic accident dataset," *Kaggle*, 2019. Available: `https://www.kaggle.com/sobhanmoosavi/us-accidents`.

[2] S. M. Moosavi and A. Betke, "Deep Accident Prediction: A deep learning framework for real-time traffic risk forecasting," *arXiv preprint arXiv:1910.01597*, 2019.

[3] Y. Gorishniy, I. Rubachev, V. Khrulkov, and A. Babenko, "Revisiting Deep Learning Models for Tabular Data," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.