Machine Learning Approaches for News Analysis ICS5110: Applied Machine Learning

Samuel Darmanin
Michelle Emma Desira
Jamal Muhsen
Leah Vella
Faculty of Information and Communication Technology
University of Malta
Malta

Abstract

Traffic accident monitoring is essential for improving road safety, particularly in small-island contexts such as Malta, where comprehensive structured accident datasets are limited. In the absence of a centralised public accident registry, initial information regarding traffic incidents is primarily reported through official police reports and news articles. This study investigates the use of natural language processing (NLP) techniques in combination with supervised and unsupervised machine learning algorithms to extract structured insights from unstructured textual data. News articles and police reports were processed to derive temporal, spatial, and contextual features, and the resulting dataset was further enriched with external information such as historical weather conditions. The enriched dataset was subsequently used to analyse spatial and temporal patterns of traffic accidents across Malta and to evaluate the performance of multiple machine learning approaches. Random Forests and Logistic Regression were applied to classify accident severity, while Support Vector Machines were employed to assess severity classification under varying feature groupings and class imbalance conditions. In addition, the K-Means clustering algorithm was used to identify potential traffic accident hotspots and to segment locations into low, medium, and high accident-prone areas. The results demonstrate that news-based text mining, when combined with appropriate preprocessing and evaluation strategies, can provide valuable insights to support more responsive and data-driven road safety analysis. All code and materials developed in this study are available in the accompanying GitHub repository; https://github.com/leivell/ICS5110_applied_ml_assignment_Jan2026/tree/main.

Abbreviations

Machine-Learning (ML), Random Forest (RF), Logistic Regression (LR), Support Vector Machine (SVM), K-Means Clustering (KMC), Regular Expressions (RegEX), Natural Language Processing (NLP), Natural Language Toolkit (NLTK), Named Entity Recognition (NER), Application Programming Interface (API), Principal Component Analysis (PCA), Receiver Operating Characteristic (ROC), Area Under the Curve (AUC), False Negative (FN), False Positive (FP), True Positive (TP), True Negative (TN), Silhouette Score (SS), Davies Bouldin Index (DBI), Shapley Additive exPlanations (SHAP), World Health Organisation (WHO), National Statistics Office (NSO), Extreme Gradient Boosting (XGBoost), Nearest Neighbor Classification (NNC), Decision Jungle (DJ)

I. Introduction

A. Assignment Aim

Traffic accidents are reported in multiple sources that serve different purposes, including official police communications and online news media. Although these sources provided valuable information, the data are largely unstructured, inconsistently formatted, and often noisy. Although such information could be readily interpreted by a human reader, automated extraction can prove challenging due to the lack of standardisation. Articles vary considerably in style, level of detail, and context, some reporting real-time traffic accidents, while others referenced historical incidents, court proceedings, or enforcement activities.

B. Data Description

The data set provided for this assignment consisted of two sources: police press releases and local news articles. Law enforcement authorities issued press releases where traffic-related incidents are described using a semi-structured narrative format. These reports frequently contained explicit references to temporal and spatial information, such as dates, times, and locations, although the level of detail and consistency varied between entries.

Local news articles published by online media outlets covered a wider range of traffic-related content. Although some articles reported real-time traffic accidents, others focused on court proceedings, enforcement operations, policy discussions, or retrospective accounts of historical incidents. As these articles were collected using traffic-related news tags, content that does not directly describe traffic accidents was also included in the dataset. Consequently, accident-related information was

frequently embedded within broader narratives, increasing the complexity of automated information extraction.

The datasets were inherently heterogeneous due to variations in source, reporting style, and contextual focus, necessitating substantial preprocessing and feature extraction before ML models could be applied. This assignment focused on the extraction of structured information from unstructured text related to traffic and the evaluation of how such extracted features contributed to ML performance. Particular emphasis was placed on temporal and contextual information, including dates, incident times, and accident characteristics, which were not consistently or explicitly reported in all articles.

C. Plan of Analysis

Three supervised and one unsupervised ML models were implemented as part of this study, with each model addressing a specific analytical objective. LR and RF classifiers were applied to the task of classifying traffic-related articles according to accident severity, specifically distinguishing between fatal and non-fatal incidents. The performance of a linear model was subsequently compared against that of a non-linear ensemble approach to evaluate differences in predictive capability.

In addition the SVM classifier and the KMC algorithm were employed to analyse temporal patterns in traffic-related incidents. Moreover, the KMC was also applied in order to determine whether there exist any traffic hotspots around Malta with the aim to segment areas which are prone to more serious accidents over other areas. Incidents were categorised into broad time-of-day groups, namely morning, afternoon, and evening, with particular emphasis placed on identifying patterns associated with peak traffic and rush-hour periods. These models were used to assess whether temporal clustering could be reliably inferred from extracted time-related features.

Model performance was evaluated using standard classification metrics for the RF, LR and SVM implementations, including (i) accuracy, (ii) precision, (iii) recall, and (iv) F1-score. Whilst for the KMC implementation, we referred to two metrics which are widely used in literature to assess the performance of clustering algorithms, namely the SS and the DBI. Comparisons were conducted across models and feature sets in order to assess the contribution of extracted temporal information relative to text-only representations.

D. Scientific Questions

In accordance with the objectives outlined in the assignment brief, this study evaluates and compares three supervised ML algorithms and one unsupervised ML algorithm applied to the task of road traffic accident severity classification and locality segmentation. The analysis is guided by a set of scientific questions focused on comparative performance, feature contribution, and evaluation under real-world data constraints.

The first scientific question addresses comparative model performance:

- **SQ1:** How do different supervised learning algorithms compare in terms of their ability to classify fatalities due to traffic accidents when trained on the same preprocessed dataset?

This question is investigated through the implementation of two classification models, namely RF and LR. Each was evaluated using consistent preprocessing steps, stratified sampling, and common performance metrics. The aim of this question was to then apply it to an Emergency Medical Dispatch Situation, as explained below.

The second scientific question examines the influence of the time of day on the severity of the traffic accident:

- **SQ2:** How does the time of the day affect the severity of traffic accidents?

This question is investigated through the implementation of two models, namely SVM and KMC, each evaluated using consistent preprocessing steps and stratified sampling, but using different performance metrics which are suggested in literature for each model.

The third scientific question focuses on identifying traffic hotspots:

- **SQ3:** Can we determine any traffic hotspots in Malta where fatalities and severe injuries are to be expected?

This question is investigate through the implementation of the KMC algorithm.

Together, these questions provide a structured framework for comparing the selected ML approaches and for interpreting their results in a manner consistent with the goals and constraints of the assignment.

II. Related Work

According to a 2022 study by the Malta National Mortality Registry, drug overdoses and traffic accidents accounted for the majority of deaths among the younger age groups [1]. Furthermore, the NSO recorded an average of over 15,000 traffic accidents over the four year period 2019 - 2023 1. These numbers are likely to be even higher over the past two years since the period above includes the Covid-19 period, where traffic accidents dipped due to lockdown protocols. At a global level, WHO estimate approximately 1.19 million accident-related deaths per year, with a further 20-50 million people suffering non-fatal injuries [2].

Collectively, these statistics highlight the importance of research in this domain. In recent years, traffic accident severity prediction has emerged as a critical research area within transportation safety, with several studies employing a variety of ML techniques to address this challenge.

A. Literature search

A 2023 study by Ahmed et al. employed several models, such as RF, DJ, and XGBoost, to predict traffic accident severity in New Zealand. For this study they used the most recent NZ road traffic accident data available and concluded that RF was the best performing model with an 81% recall. Road category and number of vehicles involved were found to be the top predicting factors. [3]. Similarly, Iranitalab et. al. used NNC, SVM, RF and KMC among other techniques to analyse traffic accident data from Nebraska, which they categorised into four levels of severity. However, they concluded that NNC performed best, followed by RF and SVM. [4].

Conversely, Amiri et al. used Australian national hospitalisation and mortality data to predict pedestrian crash severity using LR, SVM, DT and XGBoost. They concluded that XGBoost was the best performing model with SHAP analysis revealing age, gender and crash location as the key predictors of accident severity [5].

A 2024 study held in Dubai used DBSCAN to conduct a geospatial analysis of accident hotspots with the aim to reveal patterns linked to urban intersections and high-speed roadways. Additionally it also used RF to identify how environmental factors influenced accident severity, as well as cluster-specific decision trees to develop actionable rules to improve outcomes in high-risk zones [6]. Meanwhile, a UK department of transport dataset containing over 122,000 instances was used to predict crash severity. This dataset included features such as vehicle count, road type, light conditions and weather. RF was the best performing model, achieving an accuracy of 99.8% [7].

Finally, a local study employing NB, SVM, and RF to analyse factors influencing accident and highlight accident hotspots was identified. In this study the author reported 98%accuracy and 95% recall using the RF model, making it the top-performing model [8].

B. Positioning of Current Work

While literature demonstrates the effectiveness of ML approaches for traffic accident severity prediction across several different contexts, several gaps still remain. Unlike the majority of previous studies that primarily used pre-structured governmental databases or standardised datasets, this work aims to confront the challenge of extracting structured information from inherently inconsistent and unstructured text articles. Furthermore, previous studies frequently incorporated post-injury findings as part of their prediction, while parts of this study, aim to focus on solely on information accessible at the time of the incident. Additionally, Malta's size and urbanisation represent very unique challenges, making studies held in much larger countries less applicable.

In summary, this work aims to tackle three interconnected challenges: extracting useful information from unstructured text, building predictive ML models and applying the result towards actionable insights. By targeting unstructured information directly, this work provides a blueprint for other teams facing similar challenges.

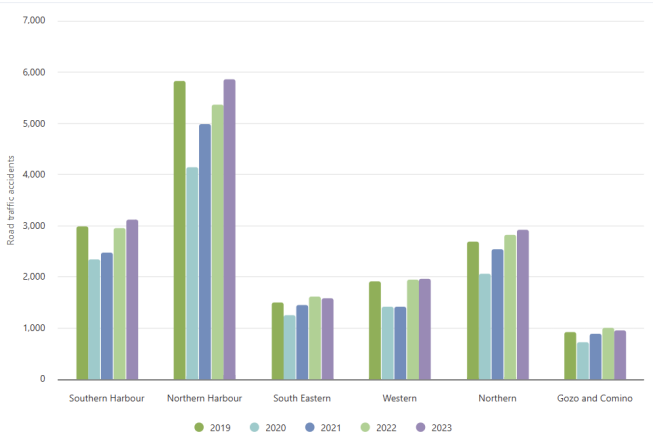

Fig. 1. Road traffic accidents by district from 2019-2023

III. Data - Feature Extraction

A. Date-of Incident Extraction

Temporal features were extracted using a conservative, hierarchical rule set to minimise incorrect inference for articles referencing incidents only contextually (e.g., retrospective reporting, court proceedings, or enforcement activity).

First, incident dates were inferred using explicit temporal keywords identified within the article content. References such as *today* or *this morning* were interpreted as occurring on the publication date, while references such as *yesterday* or *last night* were interpreted as occurring one day prior to publication.

Weekday values were inferred from the incident date whenever a valid incident date was mentioned in the content field and the weekday field remained missing. Weekdays were mapped relative to the publication date.

When both the incident date and weekday were missing, weekday terms were extracted directly from the article content using a case-insensitive RegEX pattern. Extracted weekday values were standardised to capitalised form before being stored.

Fourth, where an incident weekday was available but the incident date remained missing, the incident date was inferred by comparing the weekday of publication with the weekday referenced in the article and computing the number of days to subtract using modular arithmetic over a seven-day cycle. This approach enabled incident dates to be estimated for articles describing events relative to the publication date using weekday references.

Finally, for police-sourced entries where the incident date remained unavailable after all prior inference steps, a fallback

rule was applied in which the publication date was assigned as the incident date. This fallback was applied exclusively to police entries and only when the incident date field remained missing. Following this final inference stage, weekday inference was re-applied to ensure that weekday values were populated wherever possible.

B. Time-of-Incident Extraction

Time-of-incident extraction was implemented using a multi-stage approach in order to accommodate differences in reporting styles between police releases and news articles.

First, a RegEx pattern targeting times in the `HHMMhrs` format (e.g., `0930hrs`) was applied, primarily capturing time references commonly used in police reporting. Second, textual references such as *noon* and *midday* were mapped to the numeric time value `1200`, allowing incident times to be inferred when a precise time was not explicitly stated.

Third, a separate RegEx was used to identify `am/pm`-style time expressions frequently found in news articles (e.g., `4pm`, `4.45pm`, `4:45 p.m.`). To avoid extracting editorial update timestamps, time matches preceded by the term *updated* were ignored, and the next valid time reference was selected. Extracted `am/pm` times were subsequently normalised to a four-digit 24-hour `HHMM` format, with minutes defaulted to `00` where absent.

Time values extracted from `am/pm` formats were used exclusively to populate missing time-of-incident entries, ensuring that values obtained from police-style `hrs` reporting were not overwritten. A final formatting step was applied to enforce a consistent four-digit numeric representation while preserving leading zeros.

TABLE I
EXTRACTION RATES FOR TEMPORAL FEATURES

Source	Count	Weekday (%)	Time (%)
Police Reports	111	100.0	99.0
News Articles	321	83.4	60.0
Overall	432	87.7	69.4

TABLE II
EXTRACTION RATES FOR TEMPORAL FEATURES (ACCIDENT FLAG = 1)

Source	Count	Weekday (%)	Time (%)
Police Reports	110	99.0	98.0
News Articles	266	84.5	64.2
Overall	375	89.0	77.4

Tables I and II highlight notable differences in the completeness of temporal characteristics between police reports and news articles. Although police reports exhibit highly consistent reporting of both weekday and time-of-incident information, news articles remain substantially more variable in structure and level of detail. In particular, narrative-style reporting often omits precise temporal references or describes events indirectly, contributing to lower extraction rates. Additionally, the accident flagging process may retain a small number of articles that reference accidents only contextually rather than describing a specific incident. However, restricting the analysis to articles classified as accident-related results in a measurable improvement in temporal extraction rates, indicating that this filtering step effectively reduces noise and improves data quality for subsequent analysis.

C. Accident Location Extraction from Unstructured text

Extracting structured accident locations from unstructured accident reports posed a significant challenge. An initial investigation into the top Python packages used for NLP revealed the NLTK and spaCy as some of the best and most widely used. Both have several features, such as NER which can greatly simplify and improve text extraction [9]. However, these packages rely on pre-trained models that do not cover most of Malta's towns and cities.

As a result, a rule-based approach using RegEx was adopted. Although this approach is more customisable, it requires the creation of detailed and complex search patterns to extract relevant information. This technique involves writing detailed search patterns that are used to extract text that matches those patterns.

A comprehensive dataset containing a list of towns and cities was identified and imported into the project. It was then filtered by country to retain only Maltese localities and coordinates.

Helper functions were then created to sort the cities from longest to shortest, convert all text to lowercase, and standardise apostrophes. These are essential preprocessing steps to improve matching. Following this, a search pattern was developed to match city names within the unprocessed CSV files to the city dataset. However, after running this pattern and reviewing the results, several challenges were identified.

Challenge 1: Missing cities and naming variations. Initially, it was discovered that certain city names were not being matched. The cause was identified to be either because the name was not in the original cities dataset, or because a different variant of the name was included. As a result, several names, as well as an extra field for naming variants had to be manually added to the dataset. The latter was necessary to account for variants such as San Ġiljan and St Julian's or Pietà and Guardamanġa.

Challenge 2: Multiple locations in text. Early runs of the pattern also extracted the drivers' or victims' residences rather than the accident location. This was because most texts included several city names, and the pattern had no way to distinguish between them. Therefore, separate lists of positive and negative indicators were added. Positive indicators included words such as "on" and "Triq", and the pattern was

set to loop through these words and attempt to find a matching city within 5 words after these indicators. Conversely, negative indicators included terms such as ”resides” or ”from”. When these were encountered, the pattern was configured to skip any city names matched within 20 characters of them and instead continue searching for another valid match.

Challenge 3: Road naming conflicts Following review, it was noted that the incorrect location was sometimes identified due to road names such as Zabbar Road, Paola'. In these cases, Zabbar is the name of the road, not the locality of the accident. To address this, a further restriction was added to the pattern to ignore the city if a word such as ”road” or ”bypass” was found in the 10 characters following the city.

Challenge 4: Extraction source In some articles, a match was observed in the article title. Such matches were considered to be more accurate because when the title included a locality, this was typically the location of the accident. As a result, the code was set up to attempt to find a match in the title first and only move on to the content if there were no matches.

Unsurprisingly, since police reports were much more structured than news articles, the final pattern was able to extract the correct location from all but one of the entries. When the news articles were added, extraction fell to just over 80% of entries. While this is significantly lower, most entries that resulted in no matches were articles that did not describe traffic accidents and often had no locality mentioned. Furthermore, some failed matches were from accidents occurring abroad, which were not relevant for the scope of this study. As a result, if such entries are ignored, extraction success is expected to exceed 90%, which is quite good considering the challenges described above.

D. Text to binary flags

Given that the majority of the data provided was in text form and this text contained necessary insight for us to determine the subject of the article, we needed to parse the data in a way that we can get meaning out of text. This step was achieved by making use of Python's inbuilt *NLTK* package in order to parse the sentences in the *title* and the *content* fields. Here we had two inputs, the *title* and the *content* fields, and each had the same number of entries (1 entry per 1 news article). Each row was parsed, and any stopwords found in the English dictionary were also removed in order to eliminate unnecessary noise in the data. The words were saved to a list, and so we ended up with two lists of words.

In general, the *content* of each news article gave us much more information as to what the news article was about when compared to the *title*, however we still needed to parse the *title* field. In order to get an idea about what the majority of the new articles were about, each unique word in the list was counted and a *wordcloud* was drawn as seen in Figure 2.

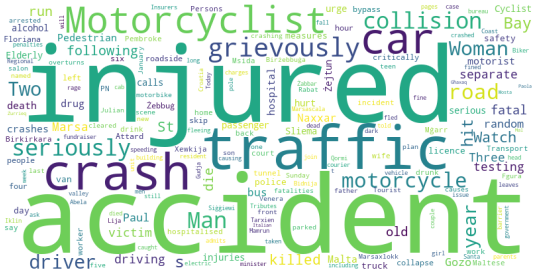

Fig. 2. Wordcloud from the Title field

The main objective of word counting was to create custom dictionaries in order for us to create new fields to our dataset, for this we used the words from the *title* field, and we managed to create 16 dictionaries. We also enriched the dictionaries with common words that are used in the same context. This was also done in order to mitigate for any words that we may have missed.

These fields were created by checking each parsed entry through a function which checks if a word in the entry matches a word in the dictionary. If the parsed entry contains at least 1 word from the dictionary then we flag it with a 1 and else we flag it with a 0. This step was repeated for each entry and for each one of the 16 dictionaries. And therefore we ended up with the following engineered fields:

- accident_flag; indicating if the entry is related to an accident
- hospital_flag; indicating if the entry is related to hospital
- fatal_flag; indicating if the entry is related to fatalities
- injury_flag; indicating if there are any injuries
- severe_flag; indicating if the entry discusses a severe injury
- minor_flag; indicating if the entry discusses a minor injury
- motorcycle_flag; checking for a motorcycle as the vehicle type
- bike_flag; checking for a bike as the vehicle type
- car_flag; checking for a car as the vehicle type
- gender_M_flag; checking for the presence of a male
- gender_F_flag; checking for the presence of a female
- heavy_vehicle_flag; indicating heavy vehicles as the vehicle type
- pedestrian_flag; indicating presence of pedestrians
- illegal_flag; indicating illegal activities/substances
- traffic_flag; indicates traffic
- control_flag; indicates that there was a loss of control whilst driving
- vehicle_count; counts all vehicle types (not from the dictionary)

After the entire dataset was created we wanted to gather some descriptive statistics on the types of vehicles that appear the

most, the type of injuries that accidents may have caused, as well as the number of accidents. This was done on both the list of words obtained from the *title* field as well as the *content* field. As we may see through Table III, the *content* field provided us with much more insight about the article.

TABLE III
DESCRIPTIVE STATISTICS ON THE TEXT TO BINARY FIELDS

Observation	Count from title	Count from content
car	68	326
motorcycle	109	234
bike	8	10
heavy vehicle	29	132
loss of control	31	208
accident	273	375
fatality	34	188
serious injury	219	375
minor injury	2	49

IV. Data Enrichment

A. Adding Weather Data

To further bolster our dataset, we decided to add relevant weather conditions. For this, the Open-Meteo API was chosen. Through the website, a list of historical weather variables was selected, including "weather code", which classifies the overall weather of the day (e.g. Rainy/Cloudy/Sunny), min, max and mean temperatures, wind speed and total rain for the day. A parameters dictionary was set up to include the aforementioned variables along with the longitude, latitude, date and timezone. The API was then set to loop through all the rows in the dataframe and extract the weather data for each location/date combination. To improve readability, weather codes were grouped and converted to text categories in a separate field.

V. Machine Learning Algorithms

Research Question 1 Overview: According to a 2022 study by the Malta National Mortality Registry, drug overdoses and traffic accidents accounted for the majority of deaths among the younger age groups [1]. Furthermore, a large-scale study analysing over two million ambulance responses to motor vehicle crashes in the United States found that longer ambulance response times were significantly associated with higher rates of mortality [10].

Emergency dispatch teams are often required to prioritise incidents under severe time pressure and on the basis of limited information obtained during the initial emergency call. Such conditions can undoubtedly lead to suboptimal resource allocation and thus delay life-saving treatment. As a result, the objective of this section of the study was to develop an ML-based framework to support decision-making, with the ultimate goal of reducing traffic-related fatalities. For this purpose, LR and RF models were implemented and evaluated to assess their effectiveness in predicting potentially fatal outcomes and to identify which model performs best in this context. The reason behind the choice of these methods was that LR would be able to provide interpretable results very easily, while RF could potentially derive unique insight from non-linear interactions that would be missed by LR. Furthermore, several cases where these models have been used successfully for this purpose were found [11] [12] [13]. To emulate real-world dispatch scenarios, the models were restricted to features that would be available to emergency dispatchers at the time of decision-making. These included accident characteristics (e.g. vehicle count, vehicle types involved), temporal factors (e.g. weather, time, day) and accident locations.

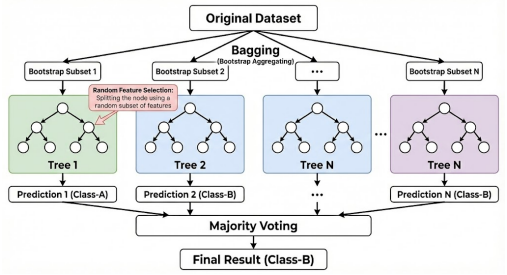

Fig. 3. Random Forest Architecture

A. Random Forests

First introduced by Leo Beriman in 2001, RFs are nowadays one of the most popular supervised learning techniques [14] [15]. As an Ensemble ML model, RFs work by constructing a large number of individual decision trees and then aggregating their results in a single, unified prediction. To improve generalisation and reduce bias, each tree in the RF is trained on a random sample of the data through a technique called bootstrapping and considers only a random subset of the features [14]. Through this randomisation, the model ensures that even the most influential features do not dominate every tree, thereby allowing it to learn more diverse patterns. For RF classifiers, as was used in this analysis, the final decision is typically dependent on the majority vote of the individual trees [16]. (Refer to Fig.3). Due to their ability to work with large, high-dimensional and imbalanced datasets, RFs have been applied to various applications, including healthcare, banking, e-commerce, social sciences and education.

Preprocessing Before running the model, several preprocessing steps were performed. As shown in Table V, this was an ongoing process with multiple iterations aimed at achieving the best predictive performance. Initially, the data was filtered using the accident_flag and fields containing irrelevant data were removed. Where feasible, numeric variables with missing values were imputed using the median. These variables included the time of incident and weather-related variables such as mean temperature, maximum wind speed and rain sum. Categorical variables, such as weather category, were imputed using the mode. Alternative imputation techniques, such as

distribution-based imputation were considered but ultimately not implemented due to time constraints.

A new binary feature, *weekend_flag* was created to capture potential differences between fatality risks during weekdays and weekends. Given the limited size of the dataset, *time_of_incident* was grouped into five different categories: Morning Rush, Late-Morning, Evening Rush, Evening and Night. The aim was that by having more observations per category, the model would be able to uncover more generalisable and actionable patterns. These features were subsequently one-hot encoded so that they could be used by the model. A similar approach was applied to weather conditions, which were grouped into Sunny/Clear, Cloudy, Foggy and Rain/Drizzle, and then one-hot encoded. Finally, all retrospective variables were removed. These variables contained information that would not realistically be available to dispatch teams at the time of decision-making and could therefore lead to data leakage in the model.

Considering the small dataset available, an 80/20 train-test split was adopted. This was because initial attempts using a 70/30 split resulted in frequent overfitting and unstable performance. Hyperparameter optimisation was performed using a `RandomisedSearchCV` rather than an exhaustive grid search. This choice allowed for much shorter waiting times and is supported by the literature which shows that randomly chosen trials are more efficient for hyper-parameter optimisation than trials using a whole grid search [17]. Following each run, incremental adjustments to data preprocessing, feature selection and hyperparameter optimisation were made to further refine performance.

Exploratory Data Analysis To better understand the data, data exploration using bar charts was performed. Fig. 4 shows a comparison between fatal and non-fatal accidents occurring on weekdays versus weekends. Surprisingly, the proportion of fatal accidents was nearly identical across both categories (approximately 45% fatal vs 56% non-fatal) (Fig.5. This reflects a known bias in this dataset, whereby severe accidents are heavily overrepresented because they are more likely to appear in police reports and news articles. Furthermore, data extraction problems due to multiple entries for the same accident and suboptimal keyword matching is likely to be artificially boosting the number of fatal accidents.

Conversely, fig. 6 illustrates fatality risk across different times of day. Accidents occurring during the late night/early morning hours show the highest fatality risks, while rush-hour periods show comparatively lower risk. One plausible explanation for this is that while more common, rush-hour accidents tend to involve slower-moving traffic, and thus reducing the risk of serious injury. In contrast, accidents happening in empty roads or during the night are more likely to involve high-speed collisions or impaired driving, which can lead to more severe outcomes.

Model Tuning and Improvement: The primary objective of this RF model was to identify and predict potentially fatal traffic accidents. Consequently, sensitivity (recal) was heavily prioritised over specificity. This is because false negatives (failing to identify a fatal accident) were considered substantially more costly than false positives (misreporting a fatal accident). Nevertheless, other metrics such as specificity, precision and AUC were also monitored. Ignoring these metrics would all the model to achieve superficially perfect sensitivity by simply predicting that all cases are fatal. Such a model would appear visually impressive but in reality offer no practical value.

The training-set AUC was also monitored as excessively

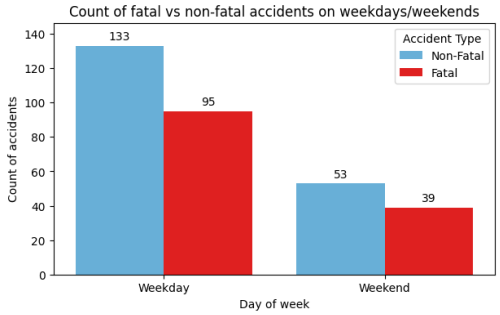

Fig. 4. Fatal and non-fatal accidents on weekdays/weekends

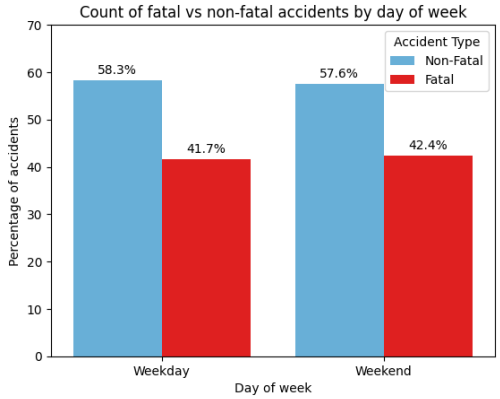

Fig. 5. Percentage of Fatal and non-fatal accidents on weekdays/weekends

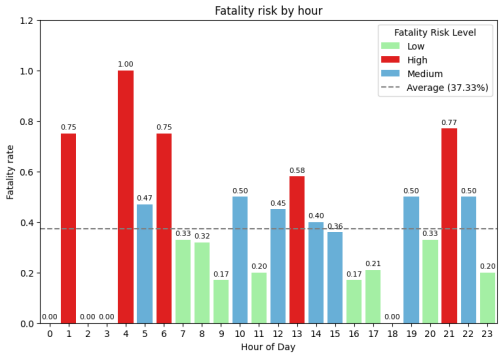

Fig. 6. Fatality risk by hour

high values (especially when compared to low testing-set values) often indicate model overfitting, thus resulting in bad generalisability and poor performance on unseen data [18].

Hyperparameter tuning was also used as a further obstacle to prevent overfitting. Initial configurations with unrestricted tree depth (`max_depth=None`), minimal samples per leaf (`min_samples_leaf=1`) and small split thresholds (`min_samples_split=2`) led to severe overfitting. Furthermore, class balancing appeared to be providing consistently better performance. As a result, the hyperparameter grid shown in Table IV was adopted.

TABLE IV
RF HYPERPARAMETER GRID

Parameter	Values	Description
bootstrap	True	Sub-sample size controlled by the max_samples.
n_estimators	[50,100,200]	No. of trees in forest
min_samples_split	[3,5,10]	Min. no. of samples to split internal node
min_samples_leaf	[2,4]	Min. no. of samples to be at a leaf node.
max_features	[”sqrt”,”log”]	Maximum no. of features used.
class_weight	[”balanced”]	Balancing for class imbalance.
max_depth”	[3,5,8,10]	Max dept of tree

TABLE V
RF PERFORMANCE ACROSS DIFFERENT FEATURE/THRESHOLD COMBINATIONS

Data	Filter	Score	Treshold	Dim.	AUC	Sens.	Prec.
1	No	Acc	0.5	(208, 30) (53, 30)	0.733	0.67	0.63
2	No	Acc	0.5	(345, 25) (87, 25)	0.751	0.84	0.67
2	AF	**Recall**	0.4	(300, 24) (75, 24)	0.711	0.94	0.58
3	**AF**	Recall	0.45	(300, 13) (75, 13)	0.555	0.68	0.52
4	**AF_C**	Recall	0.45	(256, 21) (64,21)	0.626	0.81	0.55
4	AF_C	Recall	**0.4**	(256, 21) (64,21)	0.52	0.93	0.43
4	AF_C	**Auc**	**0.35**	(256, 21) (64,21)	0.69	0.89	0.48
4	**No**	Auc	0.35	(345, 21) (87, 21)	0.606	0.79	0.48

In this table, the following apply to the data column: 1 = Full data but dropping all NaN rows, 2 = Imputed time and weather, 3 = Same as 2 but without retrospective columns. 4 = Categorised and one-hot encoded time + weather variables (no retrospective columns). For the filter column: No = No filter, AF = Accident_flag filter only, AF_C = Accident_flag and City filtering. Score refers to the model scoring. Dimensions are represented as (no. of observations, no. of features). Text highlighted in **bold** represents the changing factor from the previous model.

As shown in Table V, the baseline model trained with minimal preprocessing achieved low specificity (0.67). After filtering by `accident_flag` and imputing missing values, sensitivity increased substantially to 0.94 with an AUC of 0.71. The issue is that once the retrospective fields were removed, the specificity fell back to 0.68, indicating that the model relied heavily on information that would be unavailable at prediction time. To address this, the rows without `city_id` were removed rather than imputed, as such rows were more likely to be unrelated to traffic accidents. In fact, after applying this filtering, one-hot encoding and reducing the decision threshold to 0.4, sensitivity reached 0.93 without the use of retrospective columns. Despite this, an AUC of 0.52 was deemed insufficient. Consequently, scoring was changed to prioritise AUC while the threshold was reduced further to 0.35. The final configuration achieved a sensitivity of 0.89 with an AUC of 0.69 and was therefore selected as the best-performing model. The full performance metrics of this model are shown in table VI.

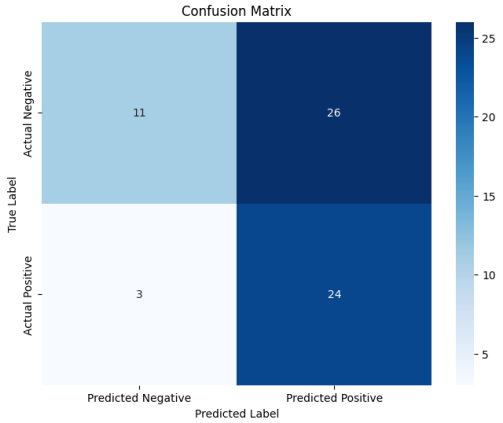

Fig. 7. RF Confusion Matrix

TABLE VI
TOP MODEL PERFORMANCE METRICS

Class	Precision	Recall	F1-Score	Support
Non-Fatal (0)	0.79	0.3	0.43	37
Fatal (1)	0.48	0.89	0.62	27

Model Evaluation: Despite extensive optimisation the final model still had several limitations. As shown in the confusion matrix plot below, 26 out of the 37 true negatives were misclassified as positive (fatal). While this level of false positives is not ideal, it was considered an acceptable trade-off given the dataset's bias towards severe accidents and the operational priority of avoiding missed fatal cases. Importantly, only 3 of the 27 potentially fatal accidents were missed, aligning with the model's primary objective.

In addition to the metrics described above, feature importance analysis was conducted on the model. As shown in Fig. 8 below, weather features such as wind speed, temperature and amount of rain were among the most influential features. Vehicle-related features, including the presence of heavy vehi-

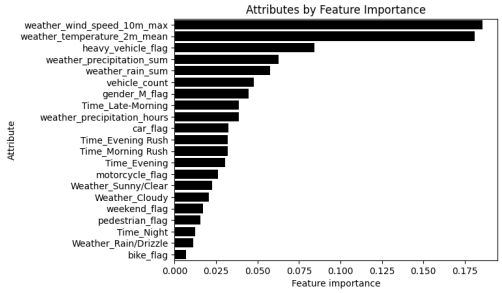

Fig. 8. RF Feature Importance Plot

cles and the number of vehicles involved, also showed a high predictive power. In contrast, factors such as rainy weather and nighttime had less importance than initially expected.

Although RF models are often classified as ”black-box” models, meaning it is hard to determine how exactly they got to the conclusion they did, SHAP plots were employed to improve explainability [19]. The SHAP analysis indicates that the involvement of heavy vehicles strongly increases predicted fatality risk, which is consistent with real-world expectations. This is because when a car collides with a heavy vehicle, the car driver is more likely to be seriously injured. High wind speeds were also observed, and low/very high mean temperatures were also associated with increased risk. Temporal factors revealed that the morning and evening periods increased risk whereas both rush hours had the opposite effect. Additionally, higher vehicle counts, weekend accidents and involvement of pedestrians were all associated with modest increases in fatality risk.

B. Logistic Regression

Preprocessing. The objective of this ML implementation was to predict fatal and non-fatal outcomes in line with **SQ1**; therefore, the task was formulated as a binary classification problem. LR was selected as an appropriate model due to its proven effectiveness in binary classification tasks. The model estimates fatality outcomes from inputs such as time, location, weather conditions, vehicle involvement, and gender indicators, enabling both prediction and analysis of the features that contribute to these outcomes.

Prior to model training, several preprocessing steps were performed to ensure reliable model performance. Fields that were not relevant for ML or that introduced unnecessary noise and complexity were removed. These included identifiers, text-based fields, selected weather variables, city information (as one-hot encoding each city would significantly increase model complexity), and post-incident severity indicators such as hospitalisation and injury flags. Severity-related variables were excluded because they are closely linked to the target variable (fatality) and could therefore result in data leakage.

Missing values for variables such as time of day, population, temperature, precipitation sum, and wind speed were imputed using the median. The median was selected as a robust

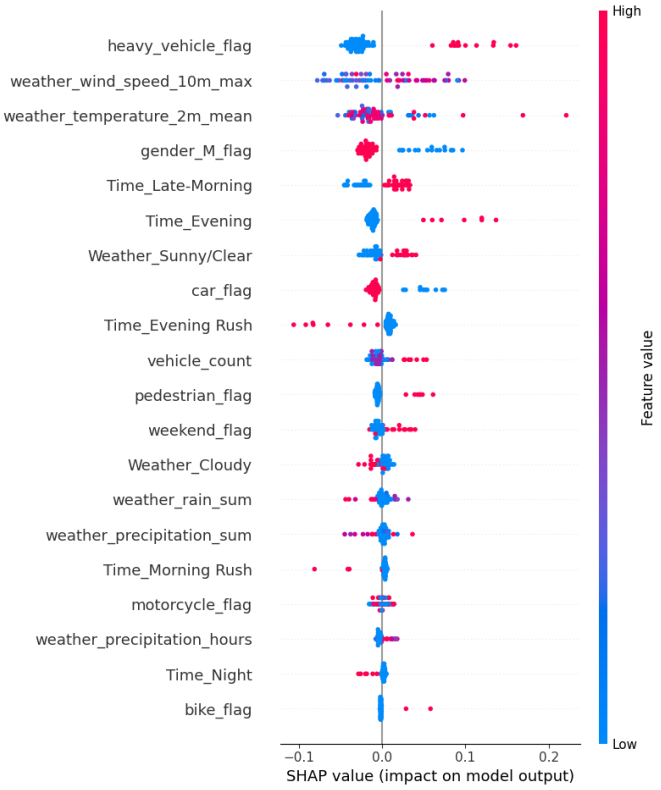

Fig. 9. RF Shap Plot

measure due to its insensitivity to outliers, thereby maintaining consistency across observations without compromising data quality. Temporal variables, such as weekdays, were encoded using binary indicator variables. This approach prevented the model from assuming an ordinal relationship between weekdays and enabled it to learn weekday-specific effects independently.

Before training, all numerical features were standardised using the `StandardScaler`. Feature scaling is particularly important for LR, as the model is sensitive to feature magnitudes. Without scaling, variables with larger numerical ranges, such as population, could disproportionately influence the model and lead to biased or unstable coefficient estimates. Standardisation ensured that all features had a mean of zero and a standard deviation of one, allowing them to contribute equally to the learning process.

Exploratory Data Analysis. To assess class imbalance, the distribution of fatal and non-fatal accidents was examined, as shown in Figure 10. The results indicate that 56% of observations correspond to non-fatal accidents, while 44% correspond to fatal accidents. This reflects a mild class imbalance that is not severe enough to hinder effective model training. Nevertheless, class weighting was applied not because of the severity of imbalance, but because of the differing costs of misclassification. In the context of traffic accidents, misclassifying a fatal accident as non-fatal (FN) carries substantially

greater real-world consequences than incorrectly predicting a fatal outcome for a non-fatal case. This adjustment increased the penalty for misclassifying fatal accidents and aimed to improve recall for fatal outcomes.

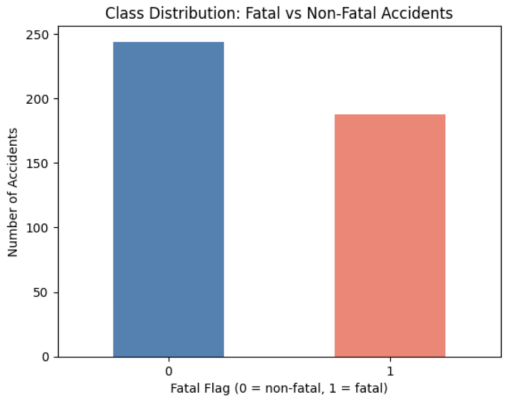

Fig. 10. Distribution of fatal and non-fatal accidents

The distribution of fatal and non-fatal accidents across weekdays is illustrated in Figure 11. Accidents were observed throughout the entire week, with no particular day showing a higher number of fatalities. Although midweek days such as Wednesday and Thursday exhibit slightly higher accident counts, weekdays alone do not appear to be a strong predictor of fatality.

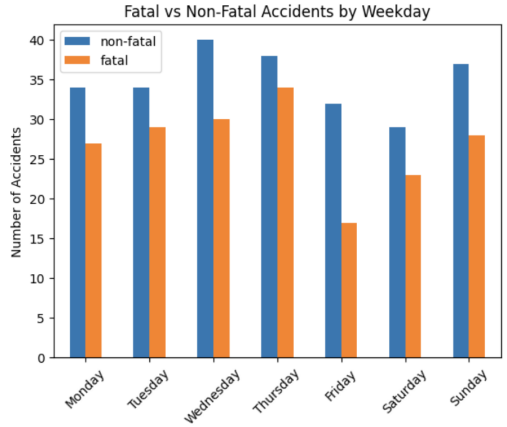

Fig. 11. Fatal and non-fatal accidents by weekday

To further explore relationships between variables, a PCA heatmap with two principal components (PC1 and PC2) was generated after feature scaling, as shown in Figure 12. Unlike the weekday analysis, which considers variables in isolation, the PCA heatmap illustrates how multiple variables interact across the dataset. The PC1 loadings indicate that variables related to vehicle involvement and loss of control load strongly onto the same component, suggesting correlated behaviour among these features. In contrast, PC2 captures additional variation through a different combination of variables, including temporal and weather-related factors, which appear more dispersed. Compared to the weekday analysis, PC1 and PC2 reveal stronger associations driven by combinations of features rather than individual variables.

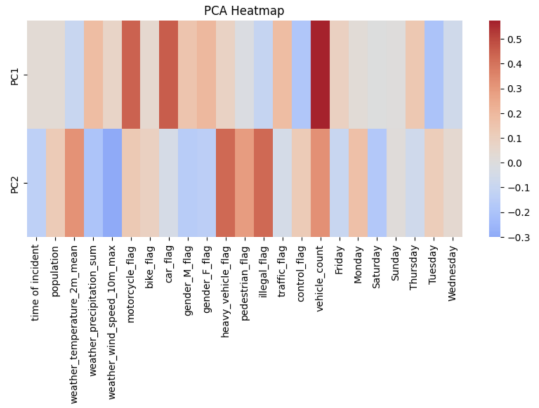

Fig. 12. PCA heatmap showing feature contributions to the first two principal components

Model Training and Hyperparameter Tuning. The dataset was split into training and test sets using a 70/30 ratio, with 70% of the data used for training and 30% reserved for evaluation. This approach allows the model to learn from a substantial portion of the data while enabling performance assessment on unseen observations. A fixed random seed was used to ensure reproducibility across multiple runs.

Hyperparameter tuning was conducted using `GridSearchCV` to optimise model performance. The regularisation parameter C, which controls the trade-off between model complexity and regularisation strength, was evaluated over a range of values $\{0.001, 0.01, 0.1, 1, 10\}$ using 5-fold cross-validation. Model selection was based on the F1-score, as this metric is particularly suitable for evaluating the correct identification of fatal accidents. The results indicated that higher values of C achieved the best performance, stabilising at $C = 1$ and $C = 10$.

Following hyperparameter optimisation, the LR model was trained on the full training dataset using the selected value of C and subsequently evaluated on the test set.

Model Evaluation and Interpretation. Model performance was assessed using multiple complementary evaluation metrics. Given the binary nature of the task and the high cost associated with misclassifying fatal accidents, particular emphasis was placed on recall, ROC analysis, and feature importance. The confusion matrix is presented in Table VII. The results show that the model correctly classified 50 non-fatal accidents and 37 fatal accidents. However, 19 fatal accidents were misclassified as non-fatal (FNs), while 24 non-fatal accidents were incorrectly predicted as fatal (FPs). Although the model successfully identifies a substantial proportion of fatal acci-

dents, the presence of FNs remains a critical concern in the context of traffic safety.

TABLE VII
CONFUSION MATRIX

	Predicted Non-Fatal (0)	**Predicted Fatal (1)**
Actual Non-Fatal (0)	50 (TN)	24 (FP)
Actual Fatal (1)	19 (FN)	37 (TP)

The classification report, shown in Table VIII, reinforces this observation. For the fatal class, the model achieved a recall of 0.66, indicating that approximately two-thirds of fatal accidents in the test set were correctly identified. Precision for fatal accidents was slightly lower at 0.61, reflecting the number of FPs. The overall accuracy of the model was 0.67, with macro-averaged precision, recall, and F1-score all at similar levels. This suggests relatively balanced performance across both classes, although recall for fatal accidents remains a notable limitation.

TABLE VIII
CLASSIFICATION PERFORMANCE

Class	**Precision**	**Recall**	**F1-score**	**Support**
Non-Fatal (0)	0.72	0.68	0.70	74
Fatal (1)	0.61	0.66	0.63	56
Overall Accuracy		0.67		

The ROC curve is shown in Figure 16, with an AUC value of **0.74**. This indicates that the model distinguishes between fatal and non-fatal accidents substantially better than random guessing. Given the complexity of real-world accident data and the limited availability of structured features, an AUC of **0.74** reflects moderate discriminative capability.

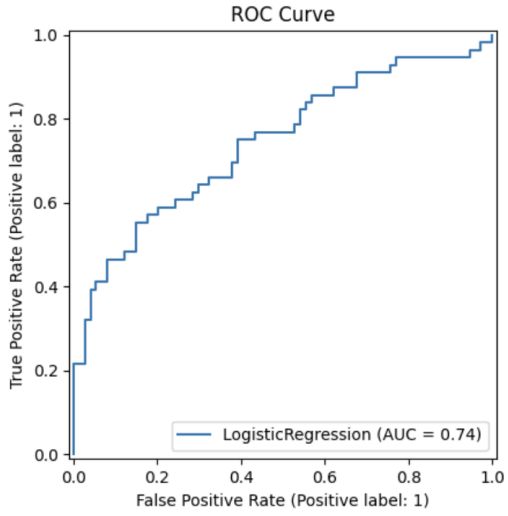

Fig. 13. Receiver Operating Characteristic (ROC) curve

Feature importance derived from the LR coefficients is illustrated in Figure 14. The most influential positive feature was `control_flag`, indicating that loss of vehicle control is strongly associated with fatal outcomes. Other features, including `illegal_flag`, `bike_flag`, population, and vehicle count, also showed positive associations with fatality. Weekday indicators, such as Saturday and Thursday, exhibited comparatively weaker positive associations.

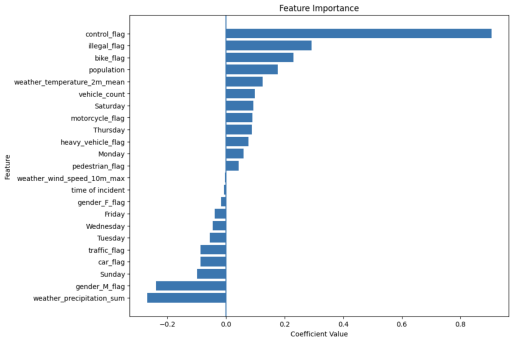

Fig. 14. Feature importance based on coefficients

Overall, the feature importance results align with real-world expectations, particularly regarding loss of vehicle control and indicators of illegal behaviour such as impaired driving. At the same time, the absence of a single dominant feature suggests that fatal accident risk arises from the interaction of multiple factors, including vehicle characteristics, traffic conditions, environmental context, and temporal elements.

Limitations and Ethical Considerations. Despite providing valuable insights, the LR model is subject to several limitations. First, the dataset is relatively small, consisting of 431 observations. In addition, the data were derived from automatically extracted information from police reports and news articles, which may contain missing values, noise, and inconsistencies. These issues introduce uncertainty and may affect predictive reliability.

From a methodological perspective, LR assumes a linear relationship between input features and the log-odds of the outcome, limiting its ability to capture complex non-linear interactions. As a result, certain patterns within the data may not be fully represented, contributing to reduced predictive performance and FN predictions, where fatal accidents are classified as non-fatal. Given the severe consequences of such errors, the model is not suitable for real-world deployment and should be interpreted strictly as an exploratory and analytical tool.

C. Random Forest vs. Logistic Regression

Since both LR and RF were trained on the same initial dataset and designed to address the same classification task, this section compares their predictive performance to determine which model is more suitable in this context. Table IX provides a summary of the evaluation metrics for both models.

TABLE IX
PERFORMANCE METRICS OF RANDOM FOREST AND LOGISTIC REGRESSION

Performance Metric	Random Forest	Logistic Regression
Precision	0.48	0.61
Specificity	0.89	0.66
F1-score	0.62	0.63
AUC	0.69	0.74

From a quantitative perspective, LR achieved higher AUC (0.74) and greater precision (0.61) for fatal accident prediction when compared to RF (AUC = 0.69, precision = 0.48). This shows that in general, LR showed a greater discriminative ability between classes and produced fewer false positive predictions. In contrast, RF achieved a substantially higher specificity (0.89) for fatal accidents, successfully identifying the vast majority of fatal accidents, compared to the 0.66 achieved by the LR model. This behaviour reflects the aggressive modelling choices made during RF development, which prioritised minimising false negatives and maximising correct classification of fatal accidents.

These results are quite surprising considering that most preliminary research showed RF outperforming linear models such as LR [6], [7], [8]. However, careful analysis of the feature selection helps explain this discrepancy. As shown in Fig.14, the most influential features for the LR model were the `control_flag` and the `illegal_flag`, both of which were considered as retrospective features during RF development and thus deliberately excluded. When these features were included, RF performance improved substantially, achieving a sensitivity of 0.84, AUC of 0.751 and precision of 0.67, thereby outperforming the LR model. Furthermore, when these features were included, they were also among the top-predicting features for the RF model.

In terms of interpretability, LR offers a clear advantage over RF. Its model coefficients provide direct insight into the relative importance and direction of each individual predictor. RF models, by contrast, are inherently less transparent and have to rely on SHAP analysis to partially address this limitation.

Excluding the two dominant retrospective features, LR identified `bike_flag` and city `population` to be the best positive predictors. RF similarly identified the presence of bikes as a positive predicter albeit to a lesser extent. Both models associated higher mean temperatures, increased vehicle counts and the presence of heavy vehicles with an increased risk of fatality. Somewhat unexpectedely, LR found total precipitation to have a negative impact on fatality while RF was mostly inconclusive.

With regards to the overarching research question, although neither model achieved the levels of performance we were aiming for, RF would likely be better suited for a high-stakes Emergency Dispatch situation. This conclusion is primarily driven by the RF's higher specificity when restricted to non-retrospective features, thus reducing the likelihood of overlooking fatal accidents in time-critical decision-making scenarios.

D. Support Vector Machine Classification

SVMs were employed in this study as supervised learning models for accident severity classification. SVMs aim to identify a decision boundary, referred to as a hyperplane, that separates classes while maximising the margin between the closest observations of opposing classes. These observations, known as support vectors, play a critical role in determining the position and orientation of the separating hyperplane.

In real-world accident reporting data, perfect linear separability is rarely achievable due to noise, overlapping feature distributions, and heterogeneous reporting styles. Given the presence of noisy and overlapping classes, a soft-margin SVM formulation was employed, allowing limited misclassifications through regularisation controlled by the parameter C. This formulation provides a balance between margin maximisation and classification error, reducing the risk of overfitting in the presence of noisy or overlapping features [20].

Prior to model training, the dataset was restricted to data flagged as accident-related to ensure that severity classification was performed only on relevant cases. Given the pronounced class imbalance within the dataset, with severe accidents constituting the majority of observations, stratified sampling was applied when splitting the data into training and testing subsets. Stratification preserved class proportions across splits and ensured that evaluation metrics reflected genuine model performance rather than artefacts of sampling bias.

Feature preprocessing was performed using a unified pipeline to ensure consistency across all experiments. Numerical features, including vehicle counts, binary indicator flags, and temporal variables such as hour of incident, were standardised. Categorical variables, including weekday, month, region, and weather category, were encoded using one-hot representations. Missing values were handled through imputation within the preprocessing pipeline, avoiding unnecessary data loss. To further address class imbalance, class-weighted SVM training was employed, penalising misclassification of the minority class more heavily and prioritising the correct identification of severe accidents.

To assess the contribution of different feature groups, a series of SVM models were trained using multiple feature combinations. These included models using temporal features only, vehicle-related features only, and progressively richer representations combining temporal, vehicle-related, contextual, and location-based information. Each configuration was trained and evaluated using an identical preprocessing pipeline and stratified sampling strategy, enabling direct comparison across models.

Table X summarises the classification performance obtained for each feature combination. Performance was evaluated

using accuracy, precision, recall, and F1-score, with particular emphasis placed on recall and F1-score due to the imbalanced nature of the dataset. Results indicate that while temporal features alone offer limited predictive power, their inclusion alongside vehicle-related features consistently improves classification performance. Conversely, the addition of location and weather features yields minimal performance gains, likely due to data sparsity and limited variability at the geographic scale considered.

Overall, these findings demonstrate that accident severity is best characterised through a combination of participant involvement and temporal context, and that careful handling of class imbalance and sampling strategy is essential for reliable model evaluation.

TABLE X
SVM CLASSIFICATION PERFORMANCE ACROSS DIFFERENT FEATURE COMBINATIONS

Feature Set	Accuracy	Precision	Recall	F1-score
Temporal features only	0.81	0.97	0.82	0.89
Vehicle-related features only	0.73	0.95	0.75	0.84
Temporal + vehicle-related features	0.88	0.97	0.91	0.94
Temporal + vehicle + contextual features	0.88	0.95	0.92	0.93
Temporal + vehicle + location + weather features	0.91	0.95	0.95	0.95

Based on the comparative evaluation presented in Table X, the SVM model incorporating temporal, vehicle-related, contextual, and location-based features was selected for detailed error analysis. While several feature combinations achieved competitive performance, this configuration demonstrated the highest overall F1-score and recall for the severe class, indicating a strong balance between sensitivity and predictive reliability. Given the safety-critical nature of accident severity classification, prioritising recall was considered particularly important to minimise the risk of misclassifying severe incidents.

To further assess model behaviour beyond aggregate performance metrics, a confusion matrix was generated for the selected configuration. The confusion matrix provides a granular view of classification outcomes by explicitly quantifying TPs, FPs, TNs, and FNs. This representation is especially informative in the presence of pronounced class imbalance, where high accuracy alone may obscure systematic misclassification of the minority class.

By examining the distribution of classification errors, the confusion matrix highlights the model's effectiveness in correctly identifying severe accidents while illustrating the limited number of non-severe cases misclassified as severe. This analysis complements the reported precision, recall, and F1-score values, offering additional insight into the practical implications of model predictions and supporting the selection of the final feature configuration.

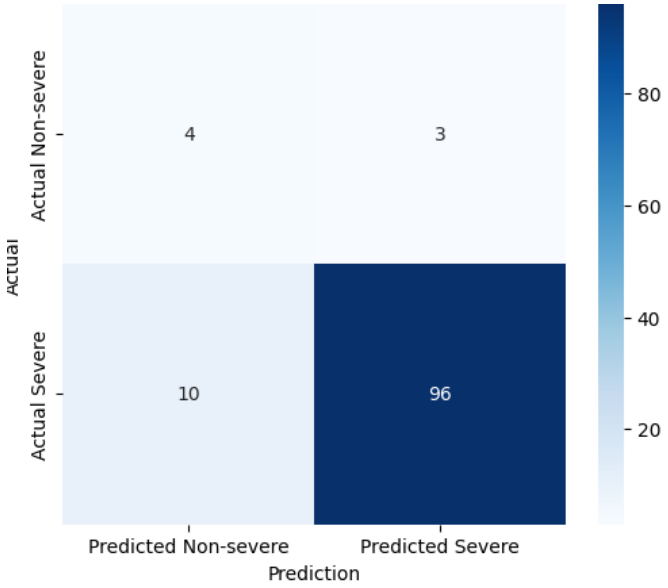

Fig. 15. Confusion Matrix for Temporal + vehicle-related features

TABLE XI
CLASSIFICATION PERFORMANCE OF THE SELECTED SVM MODEL BY CLASS

Class	Precision	Recall	F1-score	Support
Non-severe	0.29	0.57	0.38	7
Severe	0.97	0.91	0.94	106

Figure 15 and Table XI together provide a detailed view of the classification behaviour of the selected SVM model beyond aggregate performance metrics. The confusion matrix illustrates that the model correctly identifies the majority of severe accident cases, with a relatively small number of FNs, indicating strong sensitivity to severe incidents. This is particularly important in the context of road safety analysis, where failing to identify severe accidents carries greater practical risk than over-predicting severity.

At the same time, the confusion matrix reveals that a limited number of non-severe cases are misclassified as severe. This behaviour is consistent with the class-weighted training strategy adopted to address the pronounced class imbalance in the dataset, which prioritises recall for the severe class. As reflected in Table XI, this trade-off results in high precision and recall for severe accidents, while performance for non-severe cases remains comparatively lower due to the small number of available samples.

The class-specific metrics reported in Table XI further highlight this imbalance, with the severe class achieving an F1-score of **0.94** compared to **0.38** for the non-severe class. While this disparity indicates limited discrimination for non-severe outcomes, it also confirms that the model effectively captures the dominant severity patterns present in the data. Taken together, the figure and table demonstrate that the selected

feature configuration produces reliable severity predictions for accident cases while maintaining a conservative bias towards identifying severe incidents.

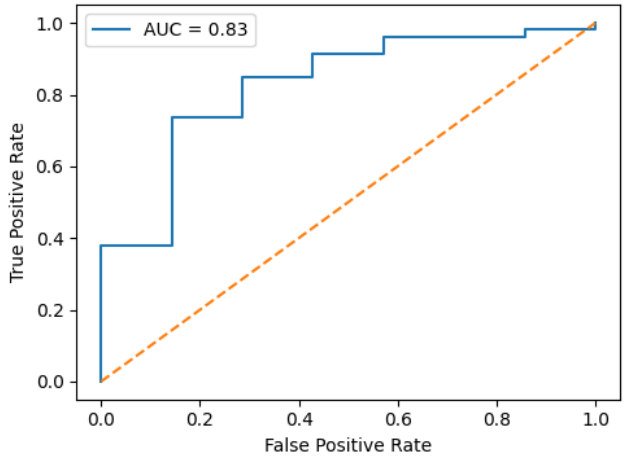

Fig. 16. ROC Curve – SVM (Time + Vehicles)

ROC analysis was used to further evaluate the discriminative performance of the selected SVM model across varying classification thresholds. Figure 16 presents the ROC curve for the model trained using temporal and vehicle-related features, with an area under the curve (AUC) of **0.83**. This value indicates strong separability between severe and non-severe accident cases, substantially exceeding chance-level performance.

The ROC curve lies consistently above the diagonal baseline, demonstrating that the model maintains a favourable trade-off between true positive rate and FP rate across thresholds rather than relying on a single operating point. This is particularly important given the pronounced class imbalance in the dataset, where accuracy alone may provide a misleading assessment of performance. The ROC analysis therefore complements the confusion matrix and class-level metrics by confirming that the model's predictive capability is robust to threshold variation.

Taken together with the confusion matrix in Figure 15 and the class-specific performance metrics reported in Table XI, the ROC results further support the selection of the final feature configuration. The combination of high AUC, strong recall for severe accidents, and stable discriminative behaviour indicates that the model effectively captures severity-related patterns while maintaining conservative and safety-oriented prediction behaviour.

Overall, the results demonstrate that while class-weighted training effectively mitigates bias towards the majority class, it cannot fully overcome the limitations imposed by extreme class imbalance and limited minority class representation. The model exhibits strong discriminative capability, as evidenced by the ROC analysis, indicating that severity-related patterns are meaningfully captured in the feature space. However, the comparatively weaker performance for non-severe cases reflects data sparsity rather than deficiencies in the modelling approach. Taken together, these findings emphasise that imbalance-aware modelling must be complemented by careful evaluation and contextual interpretation when applied to real-world accident severity data.

To provide additional intuition regarding the underlying data distribution, two-dimensional and three-dimensional visualisations of selected features were examined. These projections reveal substantial overlap between severe and non-severe accident cases, with non-severe instances frequently appearing embedded within clusters dominated by severe cases. No clear low-dimensional separation is observed, even when incorporating multiple features such as hour of incident and vehicle count.

This overlap can be partly attributed to reporting practices, as news articles are less likely to cover minor incidents unless they involve unusual circumstances or significant disruption. As a result, non-severe accidents are under-represented in the dataset and tend to share contextual characteristics with more severe cases. In addition, the local traffic environment in Malta, characterised by frequent congestion and bumper-to-bumper conditions, may further reduce the reporting of clearly non-severe outcomes. Consequently, many documented incidents exhibit similar temporal and situational features regardless of severity.

The clustering of non-severe cases within regions populated predominantly by severe cases highlights the inherent difficulty of distinguishing severity based on a limited subset of features. These visualisations therefore serve a descriptive role, illustrating the complexity and imbalance of the data rather than providing evidence of separability. The observed ambiguity in low-dimensional space reinforces the need for high-dimensional modelling approaches and supports the interpretation of the SVM results as being driven by data characteristics and reporting bias rather than limitations of the classification model

E. K-Means Clustering

KMC is a clustering algorithm to partition a dataset into *K* distinct and non-overlapping subsets, otherwise termed as clusters. The goal is to categorise data into clusters such that the points within the same cluster are more similar to each other than to those data points in the remaining *K-1* clusters. KMC is a type of unsupervised ML algorithm. Unsupervised ML algorithms learn from unlabelled data, where the algorithm attempts to identify latent patterns and structures within the dataset without any given output labels. In the scope of this study we may look at each unique article entry as a document, and our goal is to group the articles into clusters such that the articles within the same cluster share some common characteristics. Unfortunately a downfall to this approach is that KMC usually performs much better with continuous data rather than categorical data.

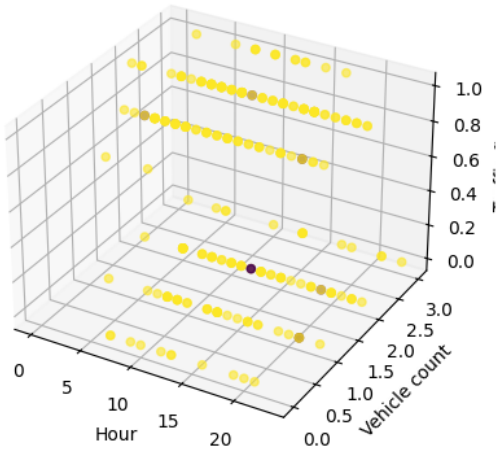

Fig. 17. Three-dimensional visualisation of accident severity as a function of hour of incident and vehicle count, highlighting class overlap and dominance of severe cases.

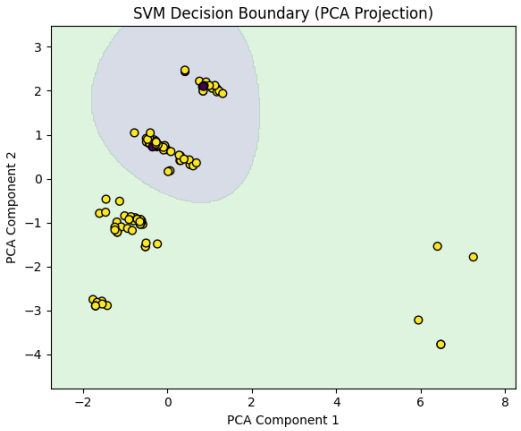

Fig. 18. Two-dimensional projection of accident severity illustrating overlap between severe and non-severe cases.

The elbow method is used to determine the optimal number of clusters needed to properly segment the dataset in the most meaningful way. It visually suggests the value of *K* (through a graph) such that having *K+1* clusters will not add any more information to help segment the dataset.

The SS is a measure of similarity of datapoints within the same cluster. It ranges between -1 and 1, with a score closer to 1 signifying well-clustered data whilst -1 implies that data points should be clustered into different clusters. A score close to 0 shows no similarity. Contrary to the SS, the DBI aims to optimise the score such that it is as small as possible; a lower score signifies better separation between clusters and subsequent compactness within clusters. For the KMC algorithm in particular a DBI value within the range 0.5-0.8 often signifies strong, well-defined clusters.

The KMC implementation for this study and **SQ3** that we aim to answer was motivated by the work done by Wahyono et. al [21], who studied the application of the KMC algorithm in order to identify traffic accident hotspots, where they focused on the city of Depok. Their input dataset consisted of sub-districts within the city, the year, the month, the total number of accidents, the number of fatalities, the number of serious and the number of minor injuries. Through the elbow method, they concluded that 3 clusters would be sufficient to segment their data in the most meaningful way. Such that sub-districts within each of these clusters were found to have;

- moderate accident rates
- low accident rates
- high accident rates

As an evaluation metric they made use of the DBI and ended up with a value of 0.896, hence concluding that the application of the KMC algorithm is effective to support the decision-making process regarding traffic accident-prone areas in Depok city.

Preprocessing. For the purpose of implementing this KMC model, we were interested in identifying latent patterns in the articles which were directly related to traffic accidents. And therefore, a subset of the data was created and it was created based on the condition that *accident_flag == '1'*, allowing us 375 articles to build the KMC implementation upon.

The subset was checked for any NA or 0 values, and we first checked for NAs specifically within the *weekday* and *city* field, which were filled with **55** *NA* values. After such rows were dropped the *time* field was checked, and this contained a further **46** empty entries. Such entries were filled with the average time found in the remaining rows. This approach was taken so that we minimise as much as possible the number of eliminated entries.

Exploratory Data Analysis. Between the two data sources, the oldest record was from the *2024-12-07* whilst the most recent was from the *2025-10-14*. After filtering for traffic accidents and further data cleansing, the data was split between the two data sources as described in Table XII.

TABLE XII
NUMBER OF DOCUMENTS FROM THE TWO DATA SOURCES

Data Source	Before Cleaning	% of Total	After Cleaning	% of Total
Police Reports	111	25.69	107	37.02
News Articles	321	74.31	182	62.98
Total	432		289	

Since *weekday*, *cities*, and *weather_category* are all strings (i.e textual data), we needed to label them in order to be able to input them into any of our ML implementations.

Tables XIII and XIV describe how the weekdays and the weather categories were labelled, as well as providing a count per type for the cleaned subset for the KMC implementation.

TABLE XIII
WEEKDAYS FOR TRAFFIC ACCIDENTS

Weekday	Tag	Number of occurrences
Monday	1	36
Tuesday	2	34
Wednesday	3	50
Thursday	4	52
Friday	5	40
Saturday	6	33
Sunday	7	44

TABLE XIV
WEATHER CATEGORIES FOR TRAFFIC ACCIDENTS

Weather Category	Tag	Number of occurrences
cloudy	1	100
sunny/clear	2	96
rain/drizzle	3	93

For the *city* feature, instead of labelling distinct cities we decided to group them according to the Maltese district which they form part of. It would be interesting to see if the clustering results are effected by these districts and to analyse whether one district is more prone to traffic accidents than another. Table XV therefore, shows the district we considered, their tag in our data and the count of cities from the cleaned subset which fell under the respective district.

TABLE XV
MALTESE DISTRICTS

Maltese District	Tag	Number of occurrences
Southern Harbour District	1	68
Northern Harbour District	2	70
South Eastern District	3	35
Western District	4	43
Northern District	5	52
Gozo and Comino District	6	21

As for the *time* at which the traffic accident was found to have occurred, Table XVI contains the most popular and least popular times of the day as well as a count of traffic accidents found during these times.

TABLE XVI
THE 5 MOST POPULAR AND LEAST POPULAR TIMES FOR TRAFFIC ACCIDENTS

Time of Day	Number of occurrences
21:00	10
13:30	9
08:00	8
18:00	8
10:15	8
05:20	1
06:00	1
05:10	1
15:37	1
16:20	1

Apart from the time of the traffic accidents, we also managed to gather statistics on the number of traffic accidents which occurred with respect to the month. In the case of the two data sources available, we may see that August 2025 had the highest number of traffic accidents whilst March 2025 had the lowest.

TABLE XVII
NUMBER OF TRAFFIC ACCIDENTS PER MONTH

Month	Year	Traffic Accidents
December	2024	26
January	2025	23
February	2025	28
March	2025	12
April	2025	25
May	2025	22
June	2025	24
July	2025	38
August	2025	**50**
September	2025	27
October	2025	14

Similar to the approach taken in the latter study, the data was also grouped according to the location of the accident. From this subset, not all fields were used for implementing the KMC; in fact only the following fields were kept:

- city
- accident
- hospital
- injury
- fatal
- severe
- minor
- regions

Model Training and Hyperparameter Tuning. From the pairplots presented (see 3d_k-means_clustering_leah_vella) there did seem to be visual clusters forming between (i) *accident* and *fatal*, (ii) *hospital* and *fatal*, (iii) *injury* and *fatal* and (iv) *severe* and *fatal*. We were interested in exploring

the relationship between the number of traffic accidents and the respective fatalities in order to answer for the question regarding traffic accident hotspots since we gather the number of accidents whilst also accounting for the number of fatalities. However, the severity of the traffic accidents would also be interesting to include in such an analysis. Therefore, 2 experiments were formulated, (i) one where we performed the KMC in 2D between the *accident* and *fatal* fields and (ii) KMC in 3D between the *severe*, *fatal* and *accident* fields.

Figures 19 and 20 show the results of the elbow method for the 2D and 3D KMC, respectively. For the former we were unsure if 3 or 4 clusters were the optimal number of clusters to represent the *accident* and *fatal* relationship, and therefore we ran the KMC on these two cluster numbers and compared results in Table XVIII. For the 3D KMC implementation, we selected 3 clusters.

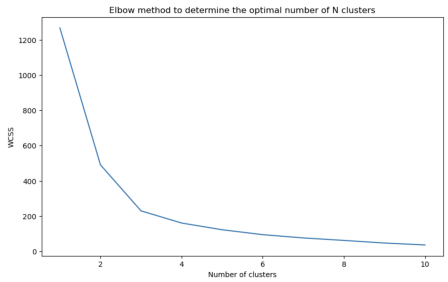

Fig. 19. Elbow Method Results for 2D K-Means

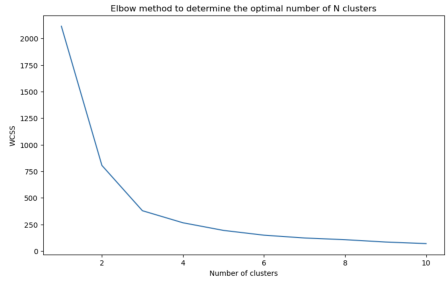

Fig. 20. Elbow Method Results for 3D K-Means

TABLE XVIII
SILHOUETTE SCORE AND DAVIES BOULDIN INDEX RESULTS

Fields	Number of Clusters	Silhouette Score	Davies Bouldin Index
accidents,fatality	3	0.5803	0.5583
accidents,fatality	4	0.5151	0.7098
accidents,fatality,severe	3	0.5838	0.5488

From Table XVIII, it is clear that the optimal setup in terms of optimising both the SS and the DBI is the 3D KMC clustering on 3 clusters, and therefore, we focus on presenting results related to this clustering setup.

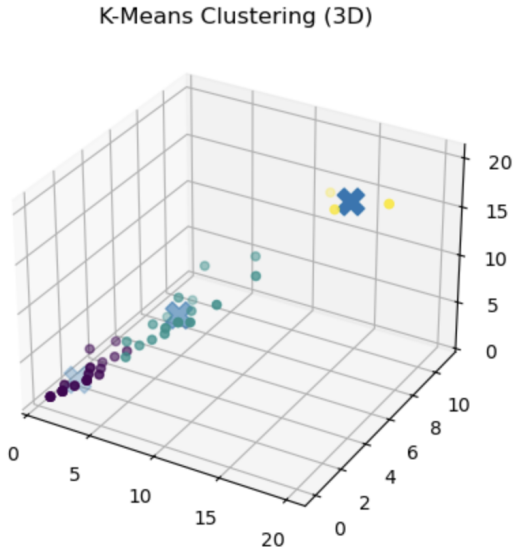

Fig. 21. 3D K-Means on 3 Clusters

Model Evaluation and Interpretation. Since we aggregated upon the different cities in Malta each datapoint represents a unique city, which was grouped into its respective cluster. Visually, through Figure 21 we may note that the clustering result did not give any overlapping clusters, which supports the **0.58** result of the SS. The clusters are also well defined and compact, especially for c_0 and c_2, which are the clusters with purple and yellow datapoints, respectively. Even though c_1 (blue datapoints) is not spilling into its neighbouring clusters, one may argue that it does not seem to be as dense. There are 4 datapoints from c_1 which are close in proximity to those datapoints in c_2 and are visually *separated* from the rest of the datapoints. These may explain why the DBI holds a value of **0.55** and not lower.

Table XIX provides the number of unique cities found in each of the clusters and under which Maltese district the cities are located. The largest cluster is c_0 with the majority of its cities found in the South Eastern and Gozo & Comino districts. Whilst the smallest cluster is c_2, which is spread between the Northern and Southern Harbour districts. The majority of the cities within c_1 are spread between the Southern and Northern Harbour districts.

Table XX provides statistics on each of the clusters, including the total number of, for instance traffic accidents which took place between the cities in that cluster, as well as the minimum, maximum and average numbers to be expected. Further details, such as the fatalities and severe and minor cases, are also accounted for.

TABLE XIX
DISTRICT SPREAD ON THE 3D K-MEANS

Cluster Code	Number of Cities	1	2	3	4	5	6
c_0	33	3	5	9	4	3	9
c_1	19	6	7	1	4	1	0
c_2	3	1	0	0	0	2	0

TABLE XX
STATISTICS FOR AGGREGATED FIELD RESULTS ON THE 3D K-MEANS

Field	Cluster Code	Minimum	Maximum	Average	Total
traffic accidents	c_0	1	5	2.58	85
	c_1	5	12	8.00	152
	c_2	14	20	17.33	52
fatalities	c_0	0	3	0.67	22
	c_1	1	7	3.47	66
	c_2	7	11	8.33	25
severe cases	c_0	1	5	2.49	82
	c_1	5	12	7.58	144
	c_2	12	20	16.00	48
minor cases	c_0	0	2	0.24	8
	c_1	0	3	0.79	15
	c_2	1	3	2.00	6

Comparing Tables XIX and XX and focusing on c_0, we may note that although this is the cluster with the largest amount of cities it has the lowest expected number of traffic accidents. The total number of traffic accidents was that of 85 of which 22 resulted in fatalities. The largest number of traffic accidents is coming from c_1, and comparing c_1 to c_0, it is evident that cities within c_0 may be considered *safer* than those cities found in c_1. There are 73.68% fewer cities in c_1, and yet 67 more traffic accidents took place, which amounts to 78.82% more traffic accidents in c_1 when compared to c_0. The severity of traffic accidents in c_1 was also *worse* since there were 66 fatalities and 144 severe cases, 43.43% and 94.74% respectively of the total number for traffic accidents in c_1.

Even though c_1 can not be categorised as safe as c_0, the situation in c_2 is definitely the most concerning. Despite that c_2 is made up of 3 Maltese cities, there were a total of 52 traffic accidents resulting in an average of 17 accidents per city compared to an average of 3 and 8 for c_0 and c_1, respectively. The rate of fatalities is also higher in c_2 since there were 25 fatalities which implies that 48.08% of traffic accidents ended up in fatalities.

Figure 22 serves as a spatial representation of the traffic accident hotspots as a result of the KMC implementation to answer for **SQ3**. We may segment the resultant 3 clusters as such;

- c_0; low accident rates
- c_1; moderate accident rates
- c_2; high accident rates

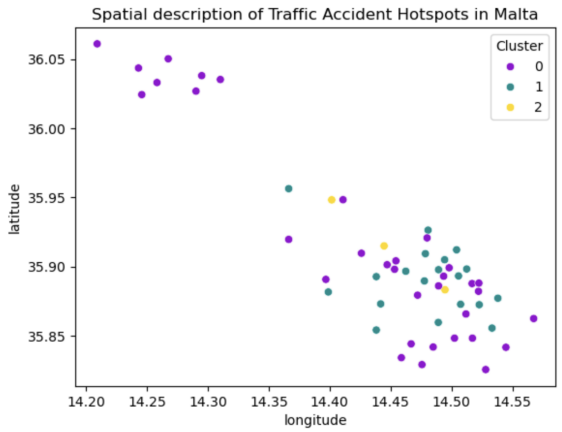

Fig. 22. A Spatial representation of the traffic accident hotspots in Malta, showing the 3 identified clusters (3D K-Means)

Figure 23 allows a visual on how the number of accidents and fatalities are spread between the 3 clusters. Supporting the results in Tables XIX and XX, c_1 contains the majority of accidents and fatalities (52.60%, 58.41%). Whilst c_0 contains more accidents than c_2 (29.41%, 17.99%), more fatalities are coming from c_2 (22.12%, 19.47%). Visuals on how the total number of severe and minor cases are spread between the clusters is presented in Figure 24. Severe and minor cases are mainly coming from c_1 (52.55%, 51.72%) and then c_0 and c_1 (29.83%, 27.59% and 17.52%, 20.69%).

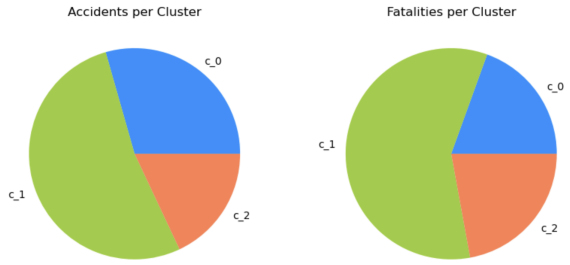

Fig. 23. Traffic Accidents and Fatalities per Cluster (3D K-Means)

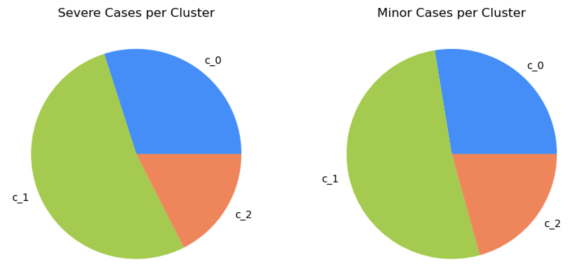

Fig. 24. Severe and Minor cases per Cluster (3D K-Means)

Support Vector Machine vs. K-Means Clustering. In order to answer for **SQ2** and for comparison purposes the KMC

algorithm was applied in order to investigate if the time of the day has an effect on the severity of a traffic accident. The traffic accident data was grouped with respect to time, and we drew clusters depending on the number of traffic accidents and the number of severe cases. Through the elbow method, the ideal number of clusters would be 3. However, it was clear that one cluster was populated by a single datapoint (representing one unique time,) which was 13:03, and this represented the average time which had been manually inserted during the data cleaning process. Therefore even though 3 clusters were drawn as shown in Figure 25, only 2 clusters gave proper insight. The SS and the DBI were **0.6985** and **0.2903** respectively. Indicating both ideal compactness within clusters and high similarity between datapoints within the same cluster.

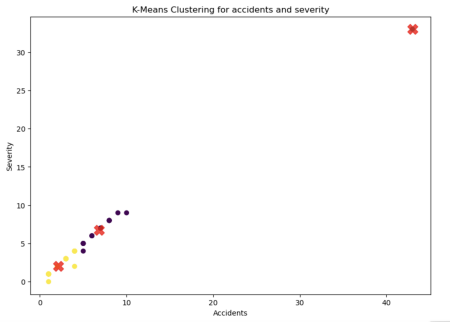

Fig. 25. 2D K-Means, grouping by time

Limitations and Ethical Considerations. There were cases where the two data sources had referred to the same traffic accident. This, therefore lead to inflation of the number of cases in specific areas. Realistically, this should have been accounted for. We also noticed that the number of traffic accidents and the number of severe cases were quite similar. The reason behind this may be two-fold; on the one hand, there may have been multiple people injured during an accident, leading to severe injuries, but also fatalities. However, on the other hand, one severe injury may have eventually led to one fatality, again inflating the number of severe cases.

VI. Limitations

The limitations discussed in this section apply across all ML models evaluated in this study, including decision tree-based, margin-based, and linear classification approaches.

A key limitation of the study is the pronounced class imbalance present in the data, with severe accidents substantially outnumbering non-severe cases. While stratified sampling and class-weighted training were employed to mitigate this issue, the limited number of non-severe samples constrains the model's ability to learn robust decision boundaries for this class. As a result, classification performance for non-severe outcomes remains unstable, as reflected in the confusion matrix and class-specific performance metrics. This imbalance also complicates the interpretation of aggregate evaluation measures, necessitating careful use of metrics such as recall, F1-score, and ROC analysis.

Another limitation relates to reporting bias within the data sources. News articles are less likely to report minor or non-severe incidents unless they involve unusual circumstances or significant disruption. Consequently, non-severe accidents are underrepresented and tend to overlap with severe cases in the feature space. This bias is further influenced by the local traffic context in Malta, where frequent congestion and dense traffic conditions may reduce the occurrence or reporting of clearly non-severe outcomes. These factors limit the extent to which severity can be distinguished based on a small number of structured features.

Additionally, the reliance on automatically extracted features from unstructured text introduces a degree of uncertainty. Although the extraction pipeline was designed conservatively to minimise incorrect inference, ambiguities in natural language reporting and inconsistent article structure may still result in missing or imprecise feature values. This is particularly evident for temporal and contextual features extracted from news articles, which are often described indirectly.

Finally, low-dimensional visualisations such as two-dimensional and three-dimensional projections provide only limited insight into the true structure of the data. As demonstrated in the analysis, substantial overlap between severity classes persists even in higher-dimensional projections. These visualisations are therefore descriptive rather than diagnostic and should not be interpreted as evidence of model inadequacy or separability.

VII. Conclusion and Future Work

This study investigated the application of multiple supervised machine learning techniques to the task of road traffic accident severity classification using structured features extracted from unstructured textual reports. A comprehensive preprocessing pipeline was developed to infer temporal, contextual, and participant-related information from heterogeneous data sources, enabling consistent model training and evaluation across all approaches. The results demonstrate that meaningful severity-related patterns can be identified despite the presence of reporting bias, noisy text data, and pronounced class imbalance.

Across the evaluated models, comparative analysis highlighted the importance of feature selection and evaluation strategy in determining model performance. Vehicle-related and temporal features consistently contributed the most predictive signal, while the inclusion of additional contextual and location-based features yielded diminishing returns due to data sparsity. Model evaluation further emphasised the necessity of using imbalance-aware metrics, as aggregate accuracy alone proved insufficient for assessing performance in this setting. The combined use of confusion matrices, class-level metrics, and ROC

analysis provided a more reliable and interpretable assessment of model behaviour.

The findings also underscore the challenges associated with real-world accident data, where non-severe cases are underrepresented and often overlap with severe cases in the feature space. Low-dimensional visualisations reinforced this observation, illustrating that ambiguity in separability is driven primarily by data characteristics and reporting practices rather than by limitations of the modelling approaches themselves. Overall, the results suggest that supervised learning models can offer valuable insights into accident severity classification when applied with appropriate preprocessing, evaluation, and contextual interpretation.

Future work could extend this study in several directions. The inclusion of additional data sources, such as official accident registries or sensor-based traffic data, may improve the representation of non-severe incidents and reduce reporting bias. Alternative resampling strategies or cost-sensitive learning approaches could further mitigate class imbalance and improve minority class performance. In addition, exploring temporal trend analysis or spatial clustering methods may provide complementary insight into accident patterns beyond severity classification. Together, these extensions would support more robust modelling and enhance the practical applicability of machine learning techniques in road safety analysis.

REFERENCES

[1] M. M. Sanchez Perez and K. England, "Annual mortality report 2022." [Online]. Available: https://dhir.gov.mt/wp-content/uploads/2024/09/Mortality-Report-2022-final-version-2.pdf

[2] Road traffic injuries. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/road-traffic-injuries

[3] S. Ahmed, M. A. Hossain, S. K. Ray, M. M. I. Bhuiyan, and S. R. Sabuj, "A study on road accident prediction and contributing factors using explainable machine learning models: Analysis and performance," vol. 19, p. 100814. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2590198223000611

[4] Iranitalab, A., & Khattak, A. (2017) - Comparison of Four Statistical and Machine Learning Methods For Crash Severity Prediction. — PDF — Support Vector Machine — Machine Learning. Scribd. [Online]. Available: https://www.scribd.com/document/940003203/Iranitalab-A-Khattak-A-2017-Comparison-of-four-statistical-and-machine-learning-methods-for-crash-severity-prediction

[5] M. A. Amiri, S. Afshari, and A. Soltani, "Machine learning approaches to traffic accident severity prediction: Addressing class imbalance," vol. 22, p. 100792. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2666827025001756

[6] M. Y. S. AlHashmi, "Using Machine Learning for Road Accident Severity Prediction and Optimal Rescue Pathways."

[7] Y. Khosravi, F. Hosseinali, and M. Adresi, "Identifying accident prone areas and factors influencing the severity of crashes using machine learning and spatial analyses," vol. 14, no. 1, p. 29836. [Online]. Available: https://www.nature.com/articles/s41598-024-81121-7

[8] D. Cassara, "Prediction of Traffic Accident Severity."

[9] P. Xiao, "Natural Language Processing," in *Artificial Intelligence Programming with Python: From Zero to Hero*. Wiley, pp. 491–542. [Online]. Available: https://ieeexplore.ieee.org/document/10951193

[10] J. P. Byrne, N. C. Mann, M. Dai, S. A. Mason, P. Karanicolas, S. Rizoli, and A. B. Nathens, "Association Between Emergency Medical Service Response Time and Motor Vehicle Crash Mortality in the United States," vol. 154, no. 4, pp. 286–293. [Online]. Available: https://doi.org/10.1001/jamasurg.2018.5097

[11] H. Khanum, A. Garg, and M. I. Faheem, "Accident severity prediction modeling for road safety using random forest algorithm: An analysis of Indian highways," vol. 12, p. 494. [Online]. Available: https://pmc.ncbi.nlm.nih.gov/articles/PMC10787871/

[12] J. Yang, S. Han, and Y. Chen, "Prediction of Traffic Accident Severity Based on Random Forest," vol. 2023, no. 1, p. 7641472. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1155/2023/7641472

[13] M. Yan and Y. Shen, "Traffic Accident Severity Prediction Based on Random Forest," vol. 14, no. 3. [Online]. Available: https://www.mdpi.com/2071-1050/14/3/1729

[14] L. Breiman, "Random Forests," vol. 45, no. 1, pp. 5–32. [Online]. Available: https://doi.org/10.1023/A:1010933404324

[15] E. Scornet and G. Hooker, "Theory of Random Forests: A Review." [Online]. Available: https://hal.science/hal-05006431v1

[16] S. A. Ahmed Salaman Hasan, Kalakech Ali, "Random forest algorithm overview," *Babylonian Journal of Machine Learning*, vol. 2024, p. 69–79, Jun. 2024. [Online]. Available: https://journals.mesopotamian.press/index.php/BJML/article/view/417

[17] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization." [Online]. Available: https://dl.acm.org/doi/10.5555/2188385.2188395

[18] P. Charilaou and R. Battat, "Machine learning models and over-fitting considerations," vol. 28, no. 5, pp. 605–607. [Online]. Available: https://pmc.ncbi.nlm.nih.gov/articles/PMC8905023/

[19] M. Panda and S. R. Mahanta, "Explainable Artificial Intelligence for Healthcare Applications Using Random Forest Classifier with LIME and SHAP," in *Explainable, Interpretable, and Transparent AI Systems*. CRC Press.

[20] scikit-learn developers, "Support vector machines," https://scikit-learn.org/stable/modules/svm.html, 2024, accessed: Jan. 2026.

[21] H. Wahyono, H. Setiaji, T. Hartati, and N. Wiliani, "K-means clustering for identifying traffic accident hotspots in depok city," *Journal of Applied and Research Computer Science and Information Systems*, vol. 2, no. 1, pp. 159–170, Jun. 2024.