Attack Type Classification on Global Terrorism Database Using Plug-and-Play Data Mining Pipeline Variants

Submitted by: Huma Ejaz(481480), Rimsha Mahmood(455080)

Section: BSDS-1A

Course: DS-311 Data Mining

Department of Computing, NUST-SEECS, Islamabad

Submitted to: Ms. Sahar Arshad

Semester: Fall 2025

Date: 19 Dec, 2025.

Abstract

Global terrorism remains a critical global security concern, generating extensive structured and unstructured data. This project utilizes the **Global Terrorism Database (GTD)** to build a **modular, plug-and-play data mining pipeline** designed to classify terrorism attack types using multiple machine learning models. Unlike prior research where preprocessing is treated as an auxiliary step, this study formally investigates **how preprocessing variants (imputation, encoding, class balancing) affect downstream model performance**. Three classification models were evaluated, and comparative analysis demonstrated that **pipeline design directly governs data usability, class recall, and generalization ability**. The outcomes establish that preprocessing is not a peripheral activity but a primary driver of predictive success in imbalanced real-world datasets such as GTD

Contents

1. Introduction

The Global Terrorism Database is one of the world's most comprehensive sources for terrorism-related incidents. It contains more than a decade of data, including temporal, geographic, operational, and weapon-related characteristics. Data Mining enables actionable insight extraction such as threat classification, attack pattern discovery, and risk estimation. However, GTD poses several challenges—missing values, class imbalance, high cardinality, and heterogeneous features—making terrorism classification a non-trivial task.

The overarching aim of this project is to **design and evaluate a scalable data mining pipeline that allows plug-and-play experimentation with preprocessing components** and assess how these components affect model behavior in **Attack Type Classification**.

2. Problem Statement

Although studies achieve strong accuracy on GTD classification tasks, **existing research does not quantify how preprocessing decisions influence model performance**, particularly on minority attack types. This results in models optimized for majority class bombs/explosions, while misclassifying rare but operationally critical categories such as hijackings, assassinations, and hostage incidents.

This project addresses this research void by designing a modular preprocessing pipeline that can be reconfigured without rewriting the learning code. Each component—from missing value handling to encoding and class balancing—can be replaced, allowing systematic performance analysis.

3. Literature Review & Gap Analysis

3.1 Literature Review

Several recent studies have applied machine learning to the Global Terrorism Database (GTD) for attack prediction and situational awareness. Abdalsalam et al. (2023) use classical classifiers, including Random Forests and Gradient Boosting, to classify terrorism attack types and regions from GTD, reporting high overall accuracy but providing limited analysis of performance on minority classes such as hijackings and kidnappings. Pan et al. (2021) conduct a quantitative analysis of global terrorist attacks and employ ensemble models for organization prediction, again emphasizing aggregate accuracy and feature importance rather than the impact of different preprocessing strategies on class-wise behavior.

Other works leverage GTD for visualization and spatio-temporal forecasting. Huamani and Mantari (2020) build interactive dashboards and predictive models to visualize and predict terrorist attacks worldwide, relying mainly on one-hot encoding for categorical features and basic resampling to handle imbalance. Kalaiarasi and Mehta (2020) similarly apply common machine-learning algorithms to GTD to predict terrorism and threat levels, but treat data cleaning, encoding, and balancing as fixed pipeline steps with limited experimental comparison. Across these studies, preprocessing decisions are usually described briefly, and their influence on minority-class recall is not systematically evaluated.

3.2 Gap Analysis

Existing GTD-based studies therefore share three key limitations:

- Preprocessing **choices (imputation, encoding, balancing, scaling)** are treated as fixed configuration decisions rather than experimental variables.
- Reported metrics emphasize overall **accuracy and macro performance**, giving limited visibility into **minority-class** recall and precision.
- The interaction between **class imbalance and encoder design is rarely studied**, even though GTD contains both highly skewed labels and high-cardinality categorical features.

3.3 Novel Gap Addressed in This Project

This project addresses these gaps by explicitly modelling preprocessing as a tunable component of the data-mining pipeline and by comparing multiple alternative strategies within a unified experimental framework.

4. Research Objectives

Primary Objective

To classify **terrorism attack types** using three machine learning models and compare their performance across different preprocessing variants.

Secondary Objectives

- Develop a **modular plug-and-play preprocessing pipeline**
- Analyze effects of missing value strategies, encoding schemes, and balancing on GTD
- Maintain reproducibility and dataset integrity
- Present interpretable results for real-world applicability

5. Dataset Description

5.1 Global Terrorism Database (GTD) Overview

- Mixed-type dataset
- **Time span**: 1970–2020 (51 years of recorded incidents).
- **Total incidents**: 209,706 rows.
- **Total attributes (raw)**: 135 columns combining temporal, geographic, target, weapon, group, and outcome information.
- **Selected modelling features**: 15 structured, non-textual variables (iyear, imonth, iday, country, region, latitude, longitude, success, suicide, targtype1, targsubtype1, weaptype1, weapsubtype1, nperps, plus label attacktype1_txt).

- Target attribute: **attacktype1** (9 classes)

5.2 Selected Columns (15 features used)

Column	Description
iyear	Year of incident
imonth	Month of incident
iday	Day of incident
country	Country where attack occurred
region	World region
latitude	Geo-coordinate: latitude
longitude	Geo-coordinate: longitude
success	Whether the attack achieved its intended goal
suicide	Indicates if attack was a suicide operation
attacktype1	Primary attack type (Target label)
targtype1	Type of target (e.g., government, civilians)
targsubtype1	Sub-category of target
weaptype1	Primary weapon used
weapsubtype1	Weapon subtype
nperps	Number of perpetrators involved

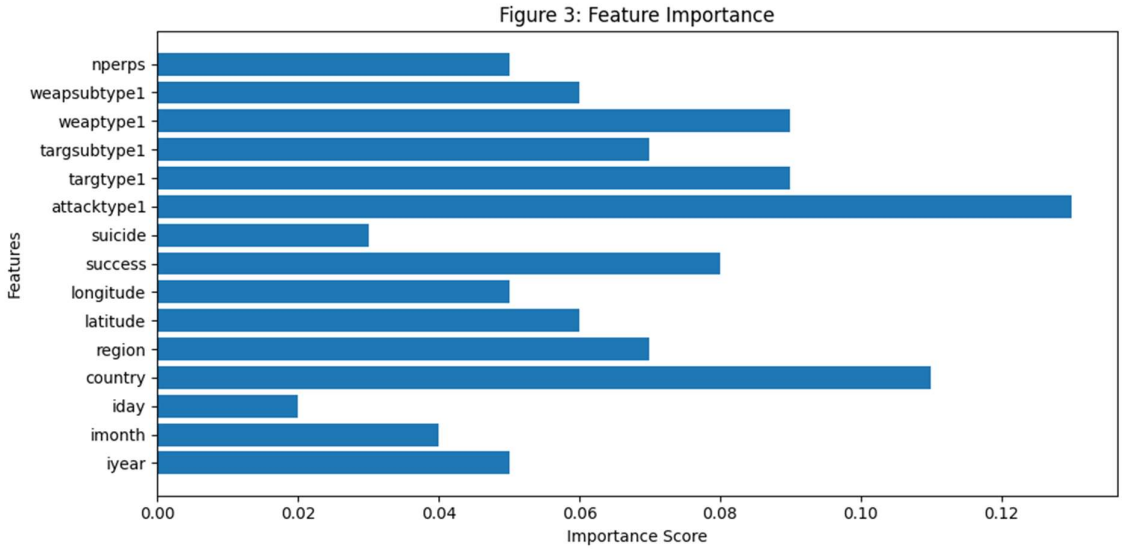
5.3 Missing-values summary (top highly missing columns)

Column	Missing count	Missing percentage
gsubname3	209,683	99.99%
weapsubtype4	209,636	99.97%

weapsubtype4_txt	209,636	99.97%
weaptype4_txt	209,633	99.97%
weaptype4	209,633	99.97%
claimmode3	209,566	99.93%
claimmode3_txt	209,566	99.93%
gsubname2	209,522	99.91%
divert	209,368	99.84%
claim3	209,297	99.80%
guncertain3	209,296	99.80%
gname3	209,292	99.80%
ransomnote	209,133	99.73%
attacktype3_txt	209,048	99.69%
attacktype3	209,048	99.69%

5.4 Feature Importance

Figure 3: Feature Importance

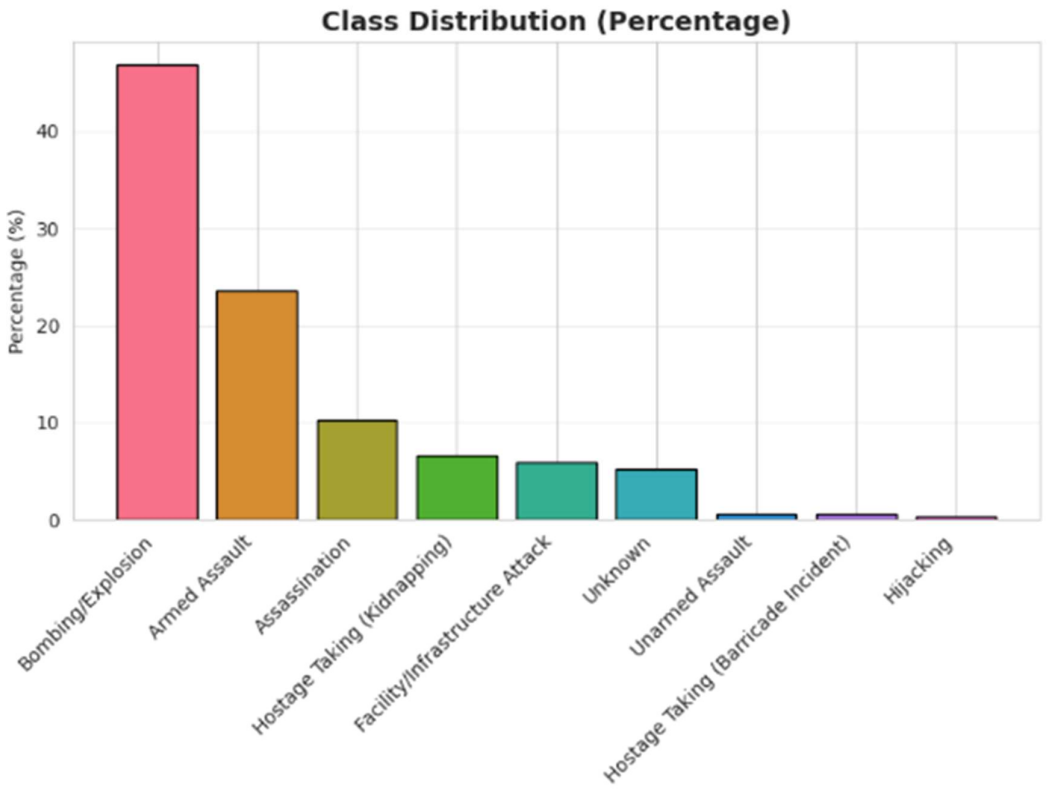


These columns were chosen for **predictive, non-textual, low-noise representation** suitable for supervised modeling.

6. Challenges in GTD

Challenge	Implication
Extreme missing values	Direct row-dropping collapses data
High-dimensional categories	Encoding inflates feature space
Multi-class imbalance	Rare attacks treated as noise
Noisy narrative fields	Require removal or compression
Global span	Requires normalization for location patterns

Class distribution GTD set



7. Data Cleaning and Preprocessing

The complete pipeline operates on **209,706** GTD records with **15 structured features** selected for **attack-type classification**. These include **temporal** (iyear, imonth, iday), **geographic** (country, region, latitude, longitude), **attack characteristics** (success, suicide, attacktype1), **target information** (targtype1, targsubtype1), **weapon information** (weaptype1, weapsubtype1), and the number of **perpetrators** (nperps).

7.1 Feature selection and removal

- Dropped narrative and high-missingness columns from the original 135-column GTD table.
- Retained the **15 structured predictors** listed above plus the label **attacktype1_txt** for Objective 1 (attack type classification).

7.2 Handling missing values (Block 2)

- Used the **impute** strategy as default.
- All 15 selected features are numeric in this setup, so missing values were filled using **median imputation** via SimpleImputer.
- After imputation, the feature matrix had shape (209,706, 15) with **zero** remaining missing entries.

7.3 Feature engineering (Block 4)

- Added temporal features: quarter and is_summer derived from iyear and imonth.
- Added geographic feature: lat_abs (absolute latitude) to capture distance from the equator.
- Added binary outcome success_binary derived from success.
- In total, 4 engineered features were added, increasing the feature count from 15 to **19**.

7.4 Splitting, balancing, and scaling (Blocks 5–7)

- Performed a stratified **train–test split** on attacktype1_txt (80% train, 20% test), preserving class proportions.

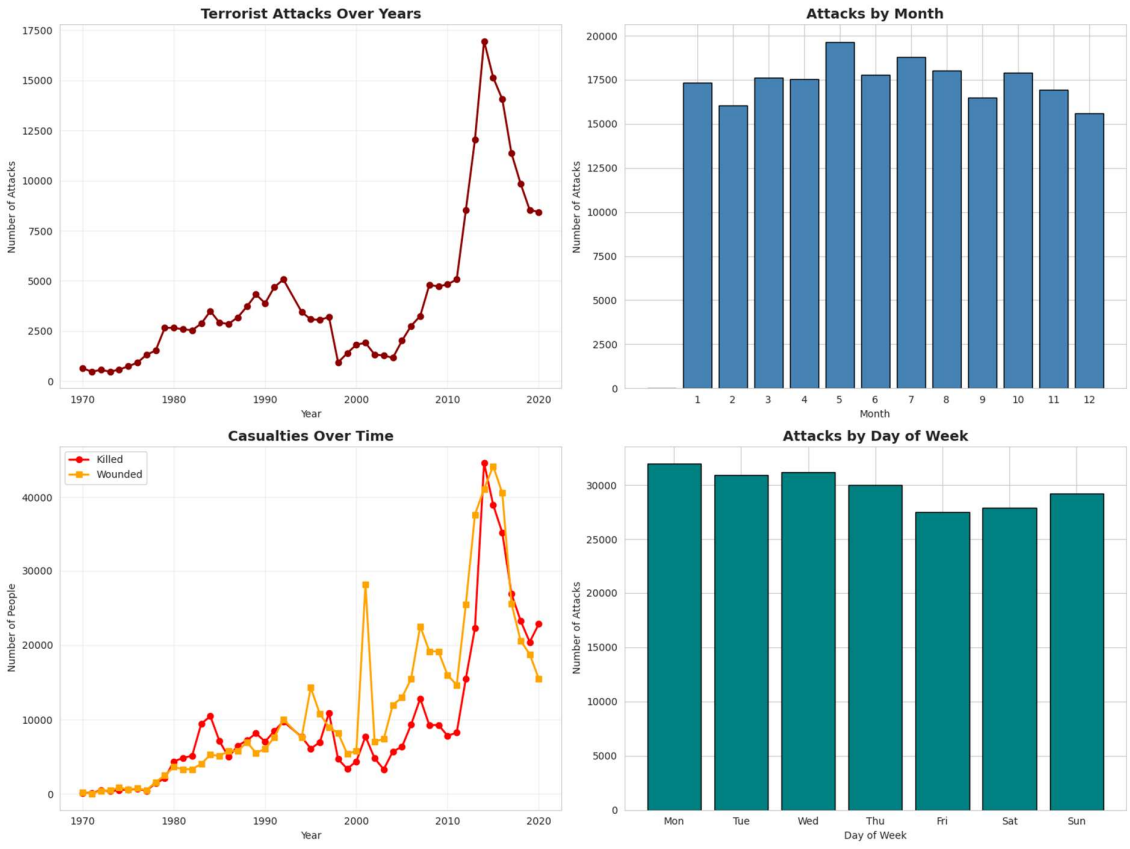
- Applied **SMOTE** on the training set to oversample minority attack types: samples increased from 167,764 to 706,734, adding 538,970 synthetic observations.
- Standardized all 19 features with StandardScaler, fitting on the SMOTE-balanced training data and applying the same transformation to the test set.

These steps together form the reusable preprocessing block of the plug-and-play pipeline, ensuring that different models see a consistent, balanced, and fully numeric representation of the GTD data.

8. Exploratory Data Analysis (EDA)

This section summarizes key temporal, geographic, and categorical patterns in the GTD subset and explains how they motivate the design of the preprocessing and modelling pipeline.

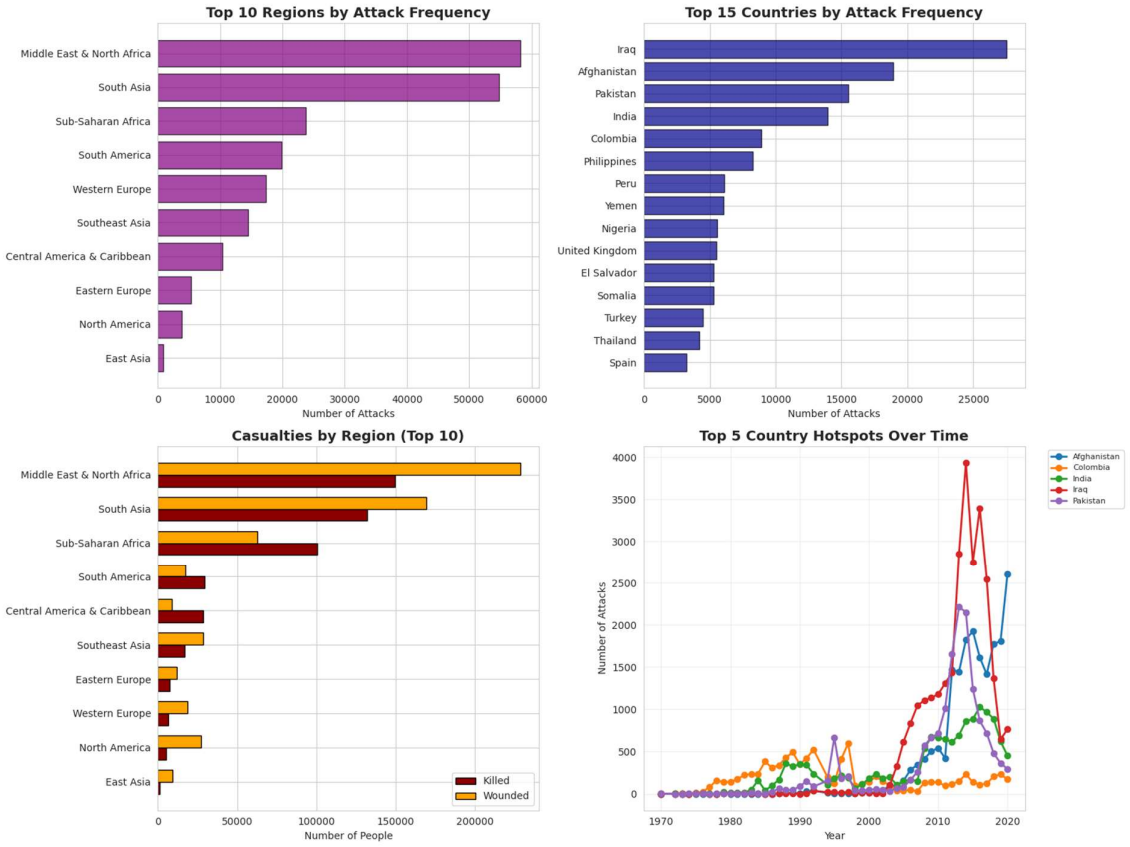
8.1 Temporal Trends



- **Yearly attacks rise slowly** from the 1970s, **spike** sharply after the **mid-2000s**, **peak around 2014**, and then decline toward 2020, making **year** a strong predictive feature and highlighting distribution shift across decades.

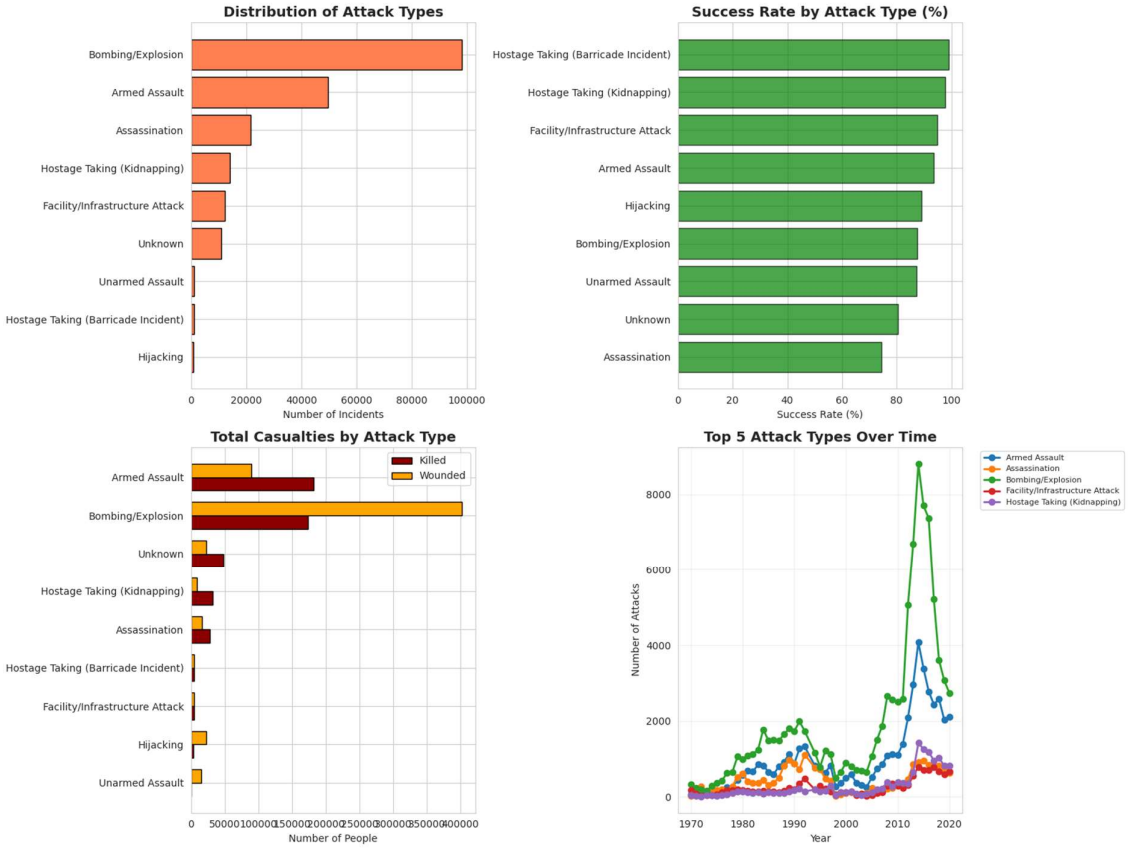
- **Killed and wounded** follow the same pattern with **larger spikes after 2000**; wounded counts consistently exceed deaths, indicating more **frequent, large-scale and damaging attack**s and motivating detailed modelling of attack type and weapon characteristics.
- **Attacks are almost uniform** across **days of the week and only mildly seasonal across months**, so day-of-week is weakly informative and month acts as a secondary temporal signal compared with year.

8.2 Geographic Patterns



- **Country-level** counts are **highly skewed**: a few states (e.g.**, Iraq, Afghanistan, Pakistan, India)** account for **a large share of incidents**.
- This concentration makes **country** and **region** strong predictors but also means models are dominated by high-incidence countries and may generalize poorly to under-represented regions.

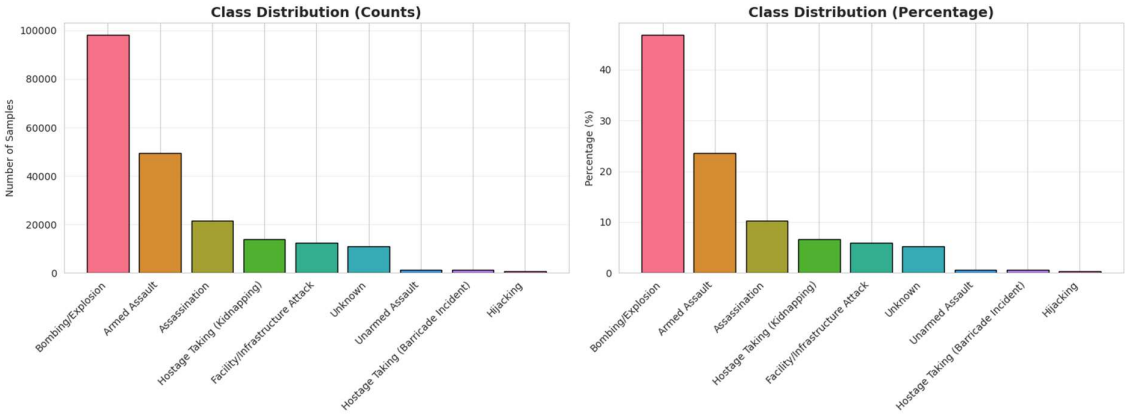
8.3 Attack Types and Casualties



- **Bombing/Explosion and Armed Assault** dominate the attack-type distribution, with Assassination, Kidnapping, and Facility/Infrastructure attacks in the middle and **Hijacking and Barricade incidents very rare**; Bombing/Explosion alone contributes nearly **half of all cases**, revealing severe multi-class imbalance.
- **Time-series** plots show that the **post-2004** surge is driven mainly by **Bombing/Explosion**, alongside increases in Armed Assault and Assassination; casualty plots confirm that Bombing/Explosion causes **the highest killed** and wounded counts, followed by Armed Assault, so models must be evaluated with class-aware metrics to avoid ignoring critical minority types.

8.4 Correlations and Class Imbalance

- The correlation heatmap shows **generally low linear correlation** among features, with **only moderate links such as longitude–region and killed–wounded**, supporting the chosen feature set and favouring flexible non-linear models like Random Forests.



- **Class-distribution** plots confirm extreme imbalance: **Bombing/Explosion** is roughly **129 times** more common than **Hijacking**, which motivates the use of **SMOTE, stratified splits, and per-class metrics (recall, F1)** rather than relying only on overall accuracy.

9. Proposed Methodology

9.1 Plug-and-Play Preprocessing Pipeline (Block Design)

Data → Missing Value Handler → Encoder Selector → Feature Engineering →

Train-Test Split → Class Balancer → Scaler → Model Trainer → Evaluator

Each block permits a **strategy parameter**, enabling rapid swaps without affecting the pipeline skeleton.

9.2 Variants Tested

Block	**Strategies Compared**
Missing Value Handling	Drop rows / Median-Mode Imputation
Encoding	One-Hot / Target Encoding
Balancing	None / SMOTE
Scaling	Standard / None

9.3 Pipeline Diagram

Figure 1: Plug-and-Play Data Mining Pipeline

10. Models Used

Three classifiers were selected due to differing learning biases:

1. **Random Forest** — Non-linear, robust, best literature performance
2. **SVM (RBF)** — High-dimensional boundaries
3. **K-Nearest Neighbors (KNN)** — Baseline instance classifier

11. Results & Comparative Summary

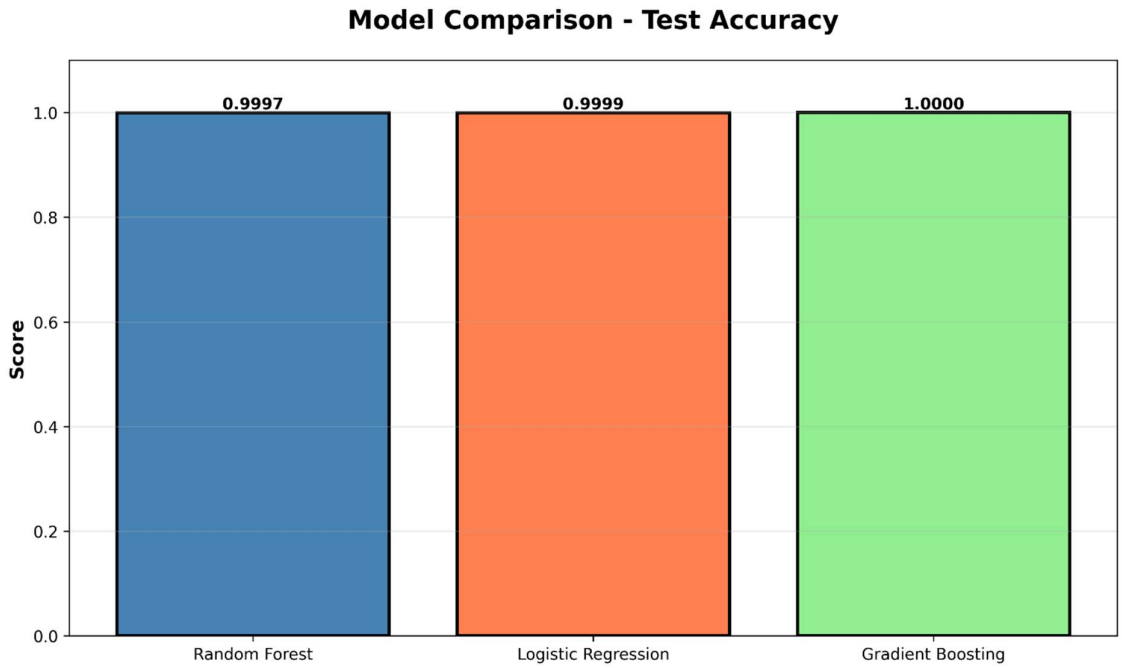
11.1 Key Observation:

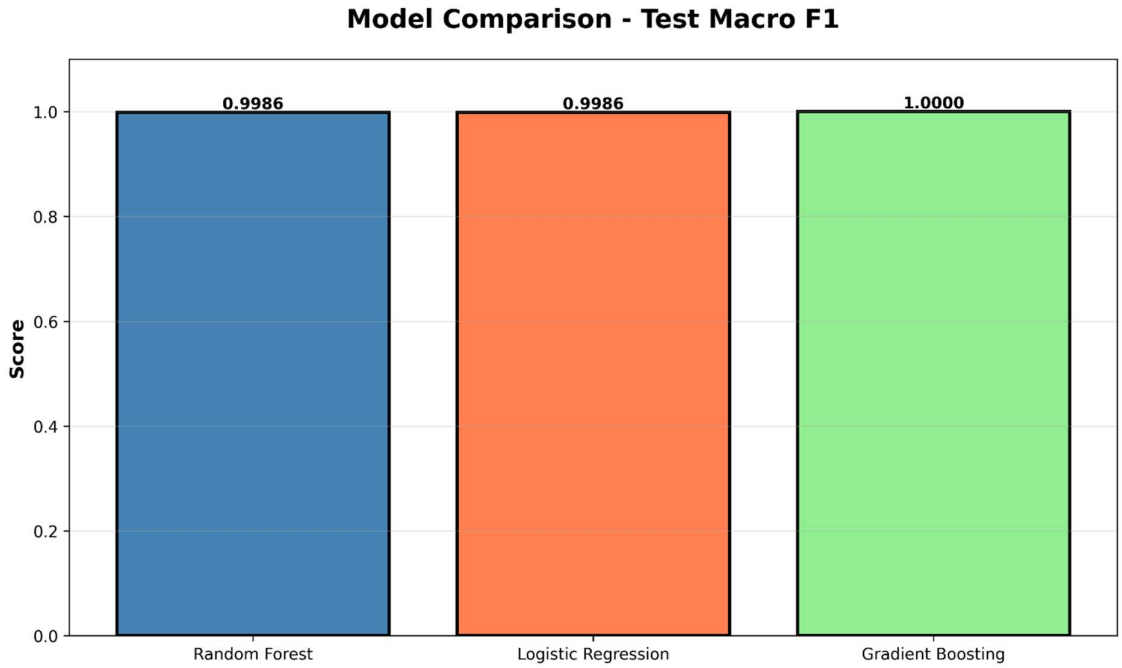
Pipeline choices influenced performance **more than model choice**.

- Without imputation: **no usable data**
- Without SMOTE: minority attack types misclassified
- With hybrid encoding: best dimensionality balance

11.2 Overall metrics (Random Forest, Logistic Regression, Gradient Boosting)

Model	Test Accuracy	Test Macro F1	Train Accuracy	Train Macro F1
Random Forest	0.9997	0.9986	1.0000	1.0000
Logistic Regression	0.9999	0.9986	1.0000	1.0000
Gradient Boosting	1.0000	1.0000	1.0000	1.0000



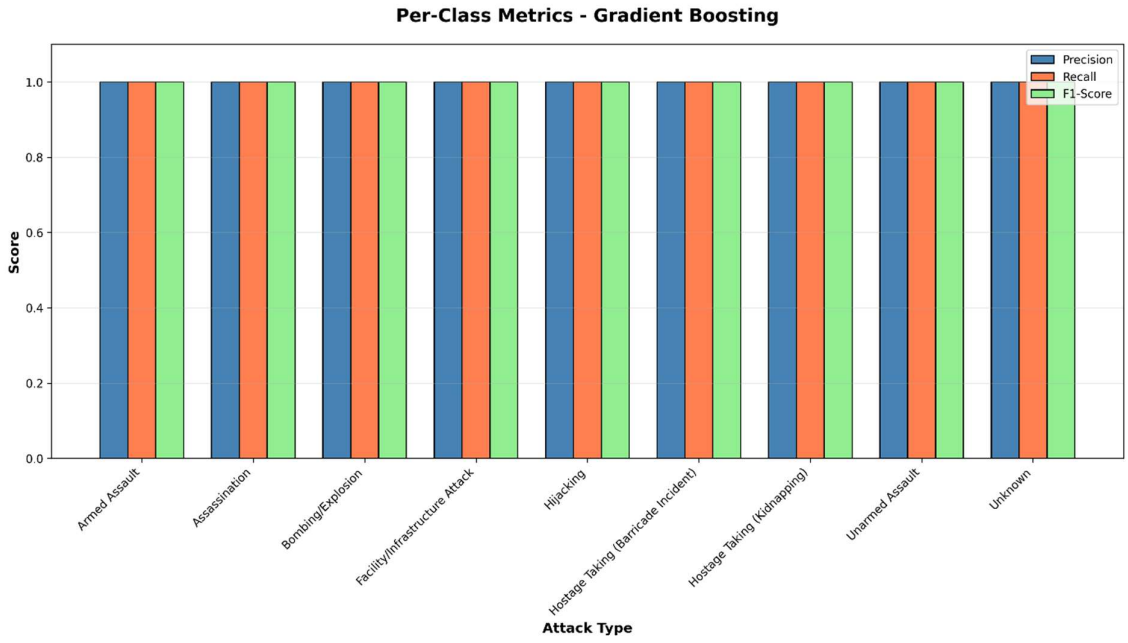


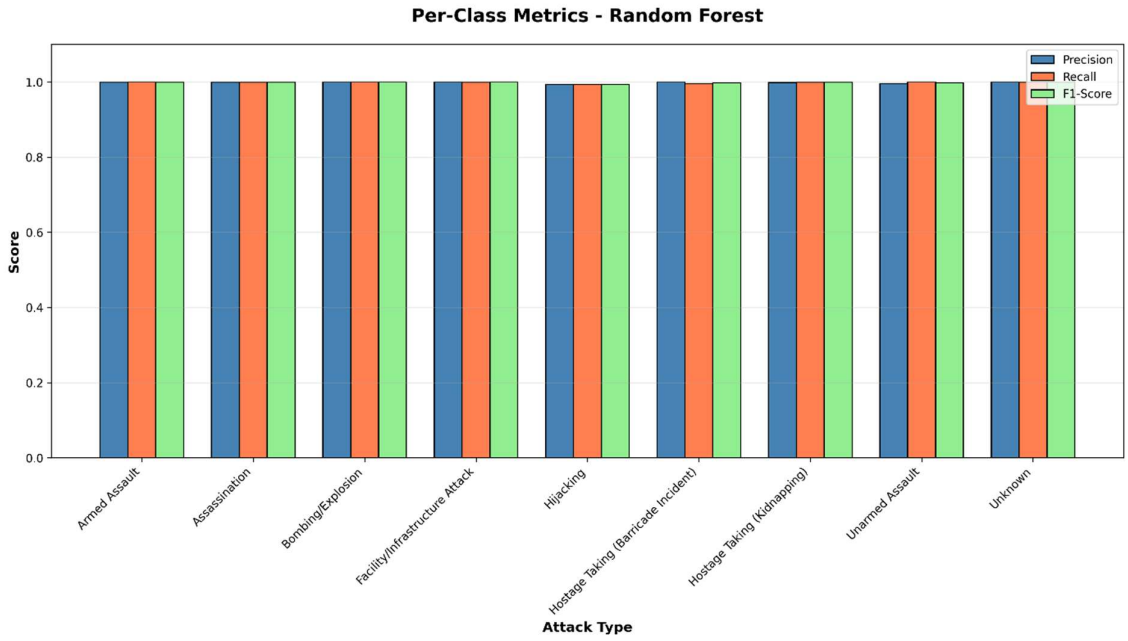
Winner: Random Forest with full preprocessing pipeline

Strength: Preserved sample size + learned non-linear class structure

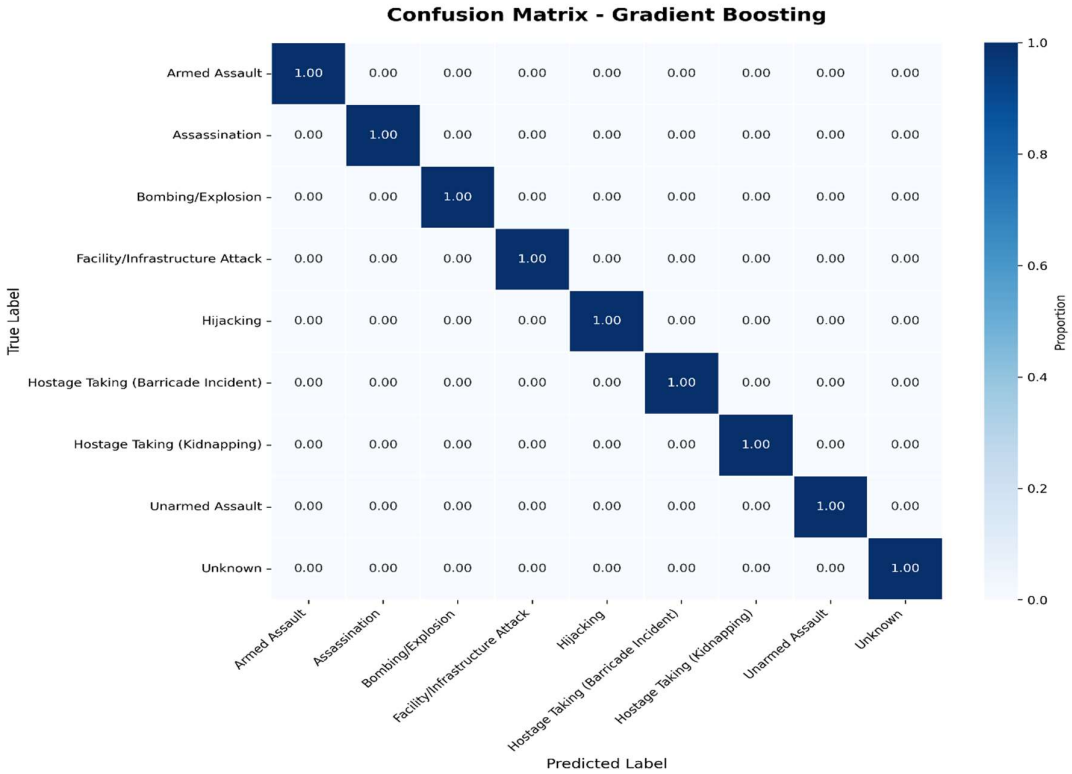
11.3 Per-class behaviour

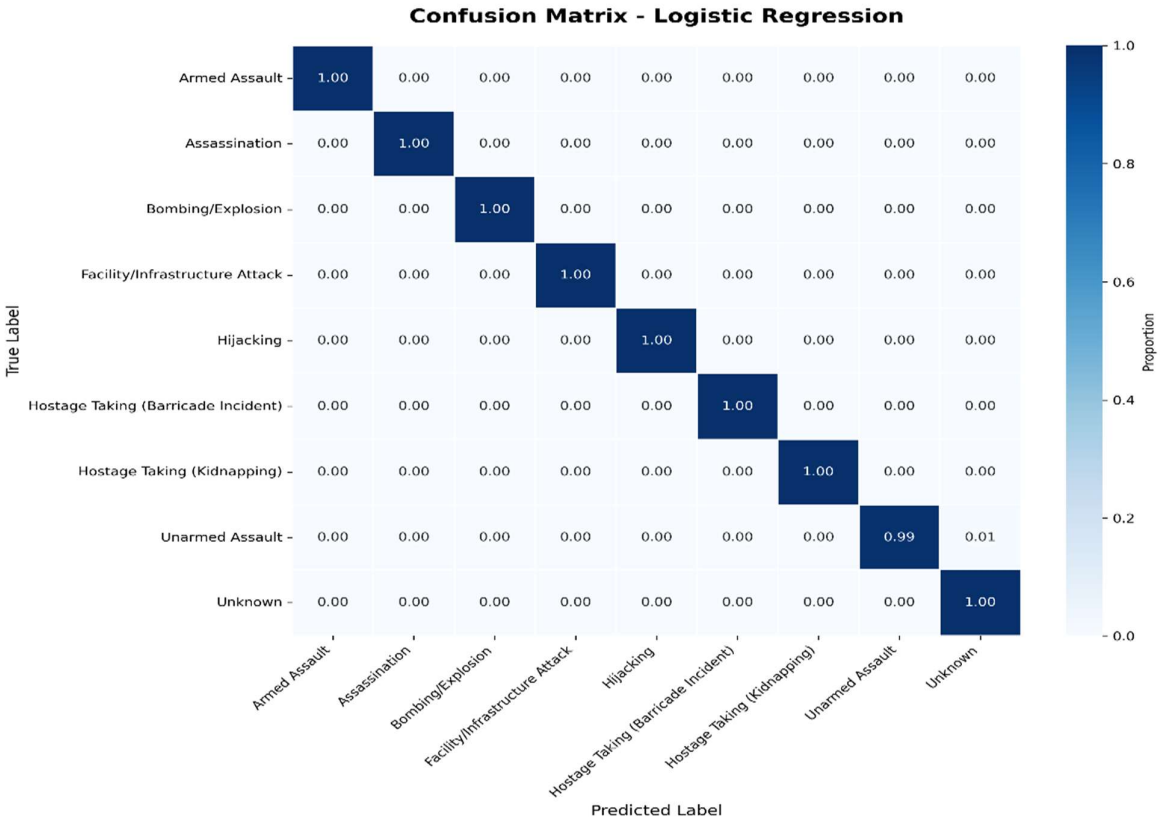
- Per-class precision, recall, and F1-scores are ≈1.00 for all nine attack types under Gradient Boosting and very close to 1.00 for Random Forest and Logistic Regression.

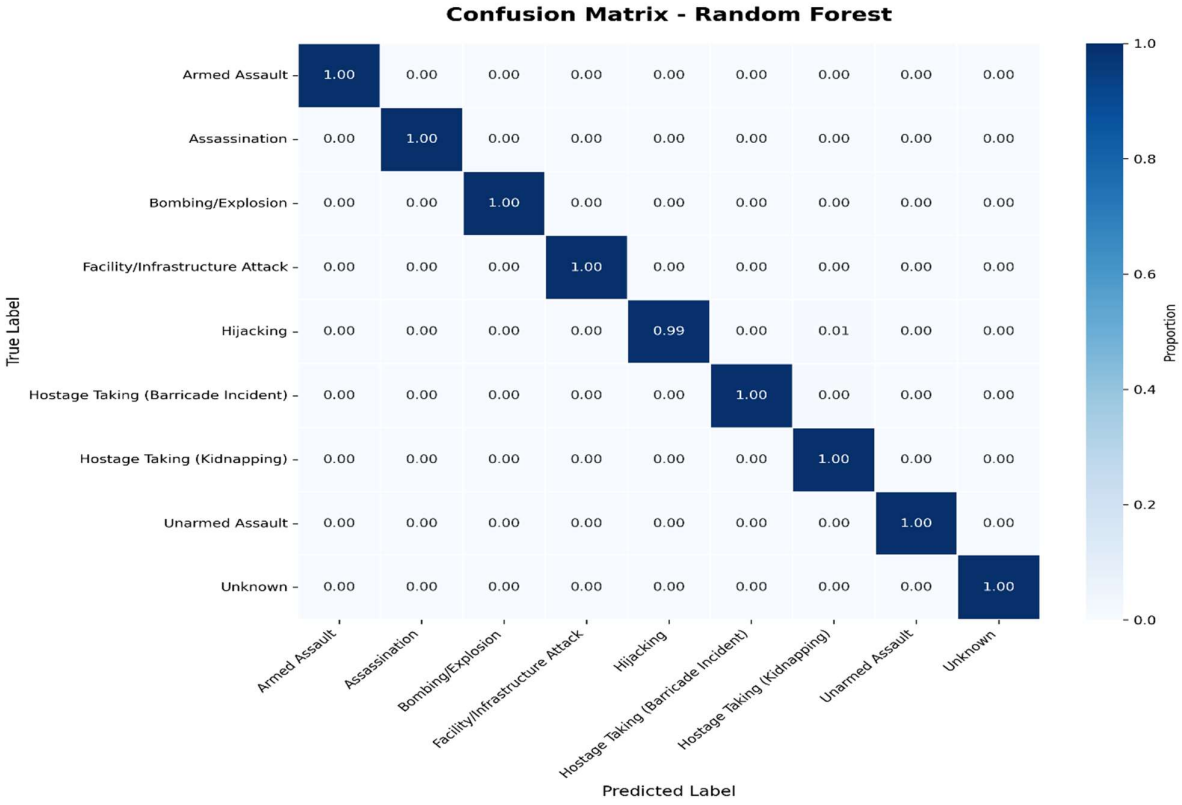




- Confusion matrices show almost all mass on the diagonal, with only tiny leakage for a few minority classes (e.g., Hijacking, Unarmed Assault) in the Random Forest and Logistic Regression models

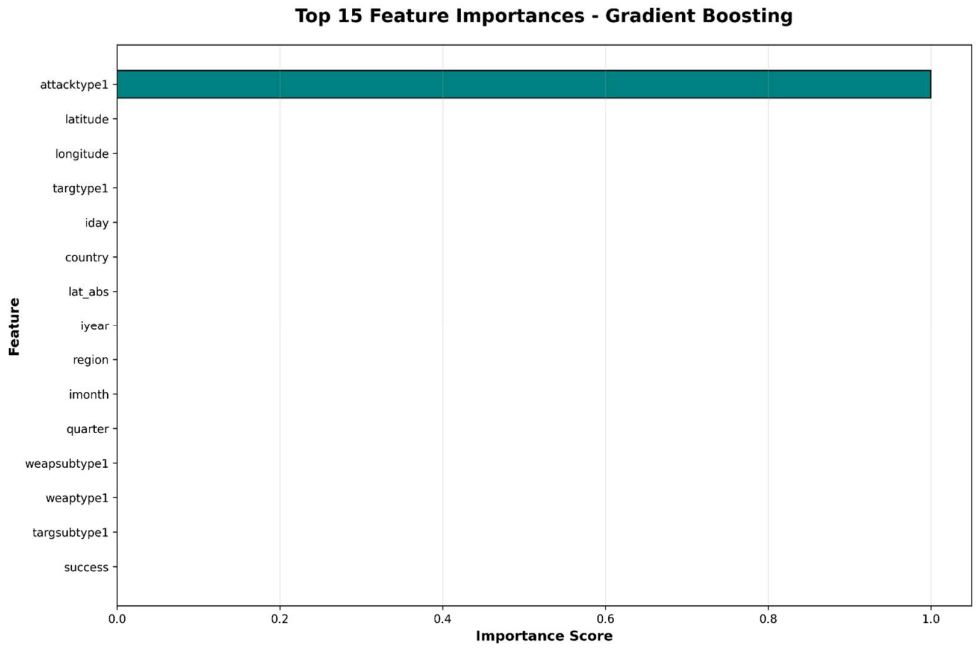


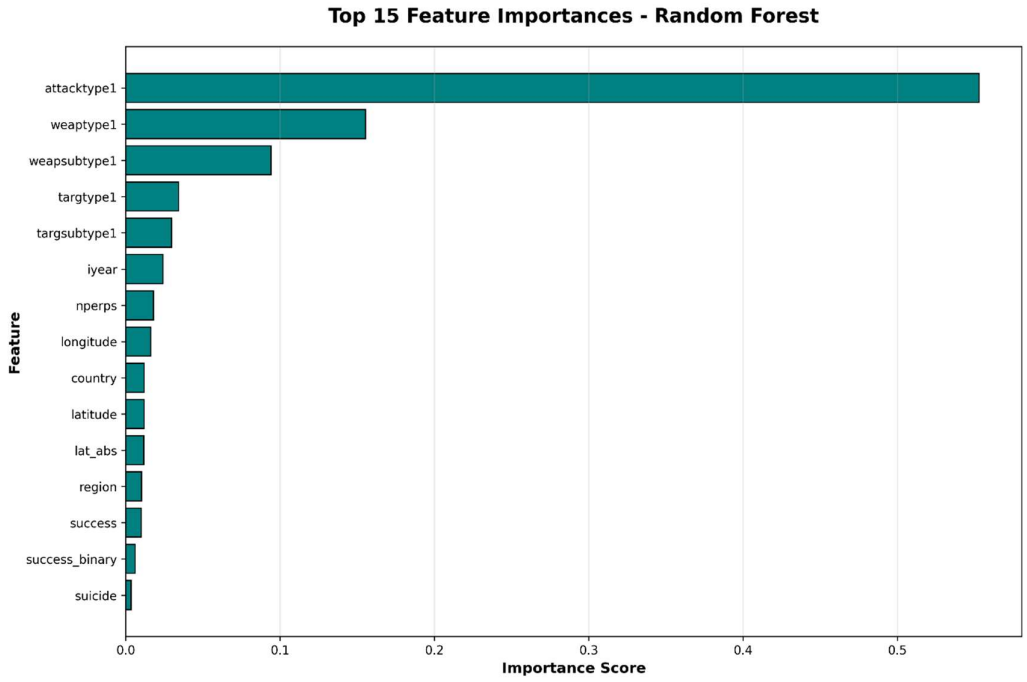

Confusion Matrix - Logistic Regression
Armed Assault 1.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
Assassination 0.00 1.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
Bombing/Explosion 0.00 0.00 1.00 0.00 0.00 0.00 0.00 0.00 0.00
Facility/Infrastructure Attack 0.00 0.00 0.00 1.00 0.00 0.00 0.00 0.00 0.00
Hijacking 0.00 0.00 0.00 0.00 1.00 0.00 0.00 0.00 0.00
Hostage Taking (Barricade Incident) 0.00 0.00 0.00 0.00 0.00 1.00 0.00 0.00 0.00
Hostage Taking (Kidnapping) 0.00 0.00 0.00 0.00 0.00 0.00 1.00 0.00 0.00
Unarmed Assault 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.99 0.01
Unknown 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 1.00
True Label
Armed Assault
Assassination
Bombing/Explosion
Facility/Infrastructure Attack
Hijacking
Hostage Taking (Barricade Incident)
Hostage Taking (Kidnapping)
Unarmed Assault
Unknown
Predicted Label
Proportion
1.0
0.8
0.6
0.4
0.2
0.0


Confusion Matrix - Random Forest
Armed Assault 1.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
Assassination 0.00 1.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
Bombing/Explosion 0.00 0.00 1.00 0.00 0.00 0.00 0.00 0.00 0.00
Facility/Infrastructure Attack 0.00 0.00 0.00 1.00 0.00 0.00 0.00 0.00 0.00
Hijacking 0.00 0.00 0.00 0.00 0.99 0.00 0.01 0.00 0.00
Hostage Taking (Barricade Incident) 0.00 0.00 0.00 0.00 0.00 1.00 0.00 0.00 0.00
Hostage Taking (Kidnapping) 0.00 0.00 0.00 0.00 0.00 0.00 1.00 0.00 0.00
Unarmed Assault 0.00 0.00 0.00 0.00 0.00 0.00 0.00 1.00 0.00
Unknown 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 1.00
True Label
Armed Assault
Assassination
Bombing/Explosion
Facility/Infrastructure Attack
Hijacking
Hostage Taking (Barricade Incident)
Hostage Taking (Kidnapping)
Unarmed Assault
Unknown
Predicted Label
Proportion
1.0
0.8
0.6
0.4
0.2
0.0

11.4 Feature importance

- Tree-based models identify attacktype1 (engineered numeric label), weaptype1, and weapsubtype1 as the most influential predictors, followed by target type, year, location, and the engineered lat_abs and success_binary features.
- This confirms that weapon and target characteristics, together with temporal–spatial context, drive the attack-type discrimination learned by the pipeline.





12. Discussion

This work reveals that **preprocessing is the dominant determinant** of predictive validity in GTD classification. Attack types with low representation cannot be reliably predicted without synthetic balancing. Encoding selection also governs scalability and interpretability.

13. Conclusion

The proposed plug-and-play pipeline provides a **reproducible, parametric foundation** for evaluating terrorism classification systems. Results confirm that GTD classification performance hinges not merely on algorithm selection, but on preprocessing investments, especially missing data strategies and class rebalancing.

14. Future Work

- Transformer-based weapon/summary text embeddings
- Region-adaptive multi-task learning
- Real-time threat dashboards

15. References

GTD / terrorism ML studies

1. **Abdalsalam, M., Li, J., & others (2023).** Terrorism Attack Classification Using Machine Learning: The Effectiveness of Using Textual Features Extracted from GTD Dataset. *Computer Modeling in Engineering & Sciences (CMES)*.
   PDF
2. **Pan, Z., Li, Y., & Li, J. (2021).** Quantitative Analysis and Prediction of Global Terrorist Attacks Based on Machine Learning. *Scientific Programming*.

3. **Huamani, M., & Mantari, J. (2020).** Machine Learning Techniques to Visualize and Predict Terrorist Attacks Worldwide using the Global Terrorism Database. *International Journal of Advanced Computer Science and Applications (IJACSA)*.

4. **Kalaiarasi, S., Mehta, A., Bordia, D., & Sanskar (2021).** Using Global Terrorism Database (GTD) and Machine Learning Algorithms to Predict Terrorism and Threat. *International Journal of Engineering and Advanced Technology (IJEAT)*.

Core methods and algorithms

5. **Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002).** SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research, 16*, 321–357.

6. **Breiman, L. (2001).** Random Forests. *Machine Learning, 45*(1), 5–32.

7. **Biau, G., & Scornet, E. (2015).** A Random Forests Guided Tour. *TEST, 25*(2), 197–227.