Applications for e-Government: Global Terrorism Analysis

by

Dimitrios Kapsis

Submitted

in partial fulfilment of the requirements for the degree of

Master of Information Systems & Services

UNIVERSITY OF PIRAEUS

February 2024

Thesis Supervisor: Prof. A. Prentza
Title: Applications for e-Government: Global Terrorism Analysis

Author: Dimitrios Kapsis.

ΣΕΛΙΔΑ ΕΓΚΥΡΟΤΗΤΑΣ

Ονοματεπώνυμο Φοιτητή: Δημήτριος Καψής

Τίτλος Μεταπτυχιακής Διπλωματικής Εργασίας: Applications for e-Government: Global Terrorism Analysis

Η παρούσα Μεταπτυχιακή Διπλωματική Εργασία υποβάλλεται ως μερική εκπλήρωση των απαιτήσεων του Προγράμματος Μεταπτυχιακών Σπουδών “Πληροφοριακά Συστήματα & Υπηρεσίες” του Τμήματος Ψηφιακών Συστημάτων του Πανεπιστημίου Πειραιώς και εγκρίθηκε στις 29/2/2024 από τα μέλη της Εξεταστικής Επιτροπής.

Εξεταστική Επιτροπή

Επιβλέπουσα (Τμήμα Ψηφιακών Συστημάτων, Πανεπιστήμιο Πειραιώς): Καθ. Ανδριάνα Πρέντζα

Μέλος Εξεταστικής Επιτροπής: Καθ. Δημοσθένης Κυριαζής

Μέλος Εξεταστικής Επιτροπής: Καθ. Μιχαήλ Φιλιππάκης

ΥΠΕΥΘΥΝΗ ΔΗΛΩΣΗ ΑΥΘΕΝΤΙΚΟΤΗΤΑΣ

Ο Δημήτριος Καψής, γνωρίζοντας τις συνέπειες της λογοκλοπής, δηλώνω υπεύθυνα ότι η παρούσα εργασία με τίτλο «Applications for e-Government: Global Terrorism Analysis», αποτελεί προϊόν αυστηρά προσωπικής εργασίας και όλες οι πηγές που έχω χρησιμοποιήσει, έχουν δηλωθεί κατάλληλα στις βιβλιογραφικές παραπομπές και αναφορές. Τα σημεία όπου έχω χρησιμοποιήσει ιδέες, κείμενο ή/και πηγές άλλων συγγραφέων, αναφέρονται ευδιάκριτα στο κείμενο με την κατάλληλη παραπομπή και η σχετική αναφορά περιλαμβάνεται στο τμήμα των βιβλιογραφικών αναφορών με πλήρη περιγραφή.

Επιπλέον δηλώνω υπεύθυνα ότι η συγκεκριμένη Μεταπτυχιακή Διπλωματική Εργασία έχει συγγραφεί από εμένα προσωπικά και δεν έχει υποβληθεί ούτε έχει αξιολογηθεί στο πλαίσιο κάποιου άλλου μεταπτυχιακού ή προπτυχιακού τίτλου σπουδών, στην Ελλάδα ή στο εξωτερικό.

Παράβαση της ανωτέρω ακαδημαϊκής μου ευθύνης αποτελεί ουσιώδη λόγο για την ανάκληση του πτυχίου μου. Σε κάθε περίπτωση, αναληθούς ή ανακριβούς δηλώσεως, υπόκειμαι στις συνέπειες που προβλέπονται τις διατάξεις που προβλέπει η Ελληνική και Κοινοτική Νομοθεσία περί πνευματικής ιδιοκτησίας.

Ο ΔΗΛΩΝ

Ονοματεπώνυμο: Δημήτριος Καψής
Αριθμός Μητρώου: ΜΕ2110

Υπογραφή:

Περίληψη

Η τρομοκρατία έχει μεγάλη επίπτωση στην κοινωνία τόσο στις ζωές των ανθρώπων όσο και στην οικονομία. Πολλές καταστροφικές τρομοκρατικές επιθέσεις έχουν συμβεί παγκοσμίως ανά τα χρόνια, προκαλώντας εκτεταμένες υλικές ζημιές αλλά και την απώλεια ανθρώπινων ζωών. Η Τεχνητή Νοημοσύνη έχει αποδειχθεί ένα πολύ πρακτικό εργαλείο για την αντιμετώπιση αυτού του κινδύνου. Σε αυτήν τη διατριβή μελετώνται δύο βάσεις δεδομένων, το "Global Terrorism Dataset" (GTD), και ένα δεύτερο σύνολο δεδομένων QFactors_Dataset, βασισμένο στο GTD, που περιλαμβάνει και ορισμένα επιπλέον χαρακτηριστικά διαφορετικής φύσης, όπως ο αριθμός των προσφύγων και μεταναστών, η μέση διάρκεια ζωής, το επίπεδο εκπαίδευσης κ.ά. Πάνω σε αυτά, δημιουργούνται και συγκρίνονται διάφορα μοντέλα Βαθιών Νευρωνικών Δικτύων (DNN) για να κατηγοριοποιήσουν τις τρομοκρατικές επιθέσεις, βάση της πολιτικής ταυτότητας που έχουν οι τρομοκρατικές ομάδες. Με αυτόν τον τρόπο εξετάζονται διάφορα χαρακτηριστικά που μπορεί να επηρεάζουν την πρόβλεψη του μοντέλου. Αρχικά, πραγματοποιείται καθαρισμός δεδομένων, και ακολουθεί η προ επεξεργασία τους για να γίνουν κατάλληλα για την εκπαίδευση των μοντέλων. Συγκεκριμένα, ήταν απαραίτητο να μετατραπούν τα λεκτικά δεδομένα (περιοχές, χώρες, τρομοκρατικές ομάδες κλπ.) σε αριθμητικά, καθώς και η κανονικοποίησή τους για να μπορέσουμε έπειτα να χρησιμοποιηθούν στα μοντέλα. Η συλλογή δεδομένων αποτελείται από εγγραφές τόσο για επιτυχημένες όσο και για αποτυχημένες επιθέσεις με χρήση διαφόρων όπλων, μεθόδων επίθεσης και στόχων. Τα μοντέλα λαμβάνουν υπόψιν διάφορους παράγοντες, συμπεριλαμβανομένων, της ημερομηνία της επίθεσης, τη χώρα, την τρομοκρατική ομάδα, τον αριθμό των θανάτων, τον τύπο της επίθεσης μαζί με τα όπλα που χρησιμοποιήθηκαν. Όλοι αυτοί οι παράγοντες αναλύονται στις επόμενες ενότητες.

Abstract

Terrorism has a great impact to the society both in the people's lives and in the economy. Many devastating terrorist attacks have occurred worldwide over the years causing extensive property damage but also leading to the loss of many human lives. Artificial Intelligence (AI) has been a very practical asset to counter this menace. In this master thesis, the Global Terrorism database (GTD) is studied, along with a second dataset QFactors_Dataset based on GTD, which includes additional features of different nature, such as the number of refugees and immigrants, average lifespan, education level, etc. Various Deep Neural Network (DNN) models are created and compared to categorize terrorist attacks based on the political identity of the terrorist groups. First, data cleaning was performed, and data preprocessing methodologies were employed to make the raw data suitable for training. More specifically, it was essential to convert our text data (regions, countries, terrorist groups etc.) to numbers, and normalize the data, in order to be able to use them in our models. The dataset consists of records for both successful and failed attacks with the

use of various weapons, attack methods and targets. The models consider various factors, including, but not limited to the date of the attack, the country, the terrorist group, the number of deaths, the type of the attack along with the weapons used. The second dataset had some additional features of different nature, like the number of refugees and immigrants, life expectancy, the education level, and others. All these factors are analysed in later sections.

Acknowledgments

I would like to thank Professor Andriana Prentza for her feedback and instructions throughout the development of my thesis.

Table of Contents

Table of Figures

List of Tables

Abbreviations

Abbreviation	Meaning
AI	Artificial Intelligence
ANN	Artificial Neural Network
ASAM	Adaptive Safety Analysis Monitoring System
CNN	Convolutional Neural Network
DM	Deep Mining
DL	Deep Learning
DT	Decision Tree
DNN	Deep Neural Network
GTD	Global Terrorism Dataset
HMM	Hidden Markov models
K-NN	k-Nearest Neighbors
LDA	Linear Discriminant Analysis
LG	Logistic Regression
LSTM	Long Short-Term Memory
MemNN	Memory Network
ML	Machine Learning
NB	Naïve Bayes
NLP	Natural Language Processing
NN	Simple Neural Network Model
NNET	Neural Network
PCA	Principal Component Analysis
QFTD	QFactors_Terrorism Dataset
RecNN	Recursive Neural Network
ReLU	Rectified Linear Unit
RNN	Recurrent Neural Network
RF	Random Forest
SMOTE	Synthetic Minority Over Sampling Technique
SVM	Support Vector Machine
STRAT	Study of Terrorism and Responses to Terrorism
URI	Uniform Resource Identifier
WNN	Wavelet Natural Network

1. Introduction

E-government is a global phenomenon and its impact to the citizens' lives, businesses, and government services, has grown importance these past few years. With the help of Data Science and Machine Learning, there has been great progress in government data analysis, classification, and prediction models, thus creating new strategies [1]. Utilizing Data Mining, we can extract information and gain insight from large volumes of data automatically by discovering patterns and association rules [2]. If we consider the vast amount of data collected from the use of the modern technological resources (internet, mobile, etc.), Machine Learning fits perfectly in assisting the new generation of e-government services, creating a safer environment for the citizens.

What is the purpose of collecting all this data? What are we trying to understand from its analysis? There are several problems that governments face while trying to assist, provide and protect their citizens. Sentiment analysis can be used to try and understand the feelings of the citizens towards the government and help in understanding citizens' direct policy suggestions [3]. With Natural Language Processing (NLP) many automations can be created [3] that interact with the users by analyzing input in many different forms like images, voice messages, or handwritten text, like chatbots [4]. Document classification is also possible and can be dealt as an Image classification problem [5]. Crime suppression is very important to ensure peoples' safety. Crime comes in many forms, e.g. both in real life and on the internet. Detection of cyber-crime can be accomplished through text analysis across various social networks [6]. Government chatbots enabled for automated crime reporting could be accessible around the clock, utilizing a crime reporting framework aided by NLP, which is capable of extracting important details from eyewitness. Iriberri and Leroy [7], emphasizing the necessity for continuous availability of crime reporting, suggested an online system for reporting crimes that employs NLP to extract relevant details from witness narratives and employs investigative interviewing techniques to pose further inquiries. Subsequently, this system consolidates all gathered information into a unified database, facilitating more organized retrieval. Such systems pave the way for more robust automated crime reporting systems in general, particularly beneficial in regions or circumstances with limited police resources. Furthermore, integrating the capability to pose relevant queries via text with text-to-speech conversion could enable automated witness interviews over voice channels.

NLP includes various technologies used to analyze large amounts of text and understand both syntactic and semantic information [8]. Software using NLP technologies will be more advantageous at such tasks as it will be able to process large amounts of text at rates greater than humans. Many of the functionalities of a government today include vast amounts of text data - from interactions with citizens to examining archives to passing orders and bylaws. NLP technologies can assist support government departments by streamlining the processing of textual information. This potentially could lead to significant improvements in the speed and efficiency of various governmental tasks. Numerous proposals and instances demonstrate how this can be accomplished across various sectors, ranging from facilitating the registration of public grievances to monitoring policy alterations [9]. Several researchers analyzed and classified government data using different DM techniques and ML algorithms, e.g. Alexopoulos et al. [10] declare that their study contributes to this research topic by offering a thorough examination of government usage of ML and Airon Zhang [11] introduced a model showcasing how Big Data can

enhance the government services by increasing the efficiency and effectiveness in the e-governance service while promoting citizen engagement in decision-making processes.

Another global problem is terrorism, where in recent decades its effects have been extremely dangerous. More and more countries have suffered from this issue, as terrorists tend to act more strategically [12]. Terrorism is characterized by intentional acts of violence and intimidation, whose main target is civilian populations to fulfill certain political, religious, or ideological goals. It involves various actions like bombings, shootings, kidnappings, and cyber-attacks, often directed at crucial locations or specific demographics to maximize its impact. Furthermore, the adaptability of the terrorist methods cannot be ignored, as they evolve to keep up with the new technologies and shifts in the global sociopolitical landscape. This adaptability poses an ongoing and complex challenge, requiring continuous efforts to address emerging threats effectively. There is no denial, that while facing these terrorist attacks, there is no security, neither stability for the countries being threatened. This has great social, economic, and political effect and leads to the decay of the society [13]. E-government has shown efforts to utilize new technological means and find new methods to counter these attacks.

Terrorist activities are typically evaluated and quantified based on the frequency of incidents and the resulting casualties, reflecting their tendencies and impact [14]. This thesis focuses on employing AI and ML to investigate the possibility of mitigating the effects of terrorism. The impact of such an application to limit the spread of terrorism cannot be ignored. Utilizing these techniques can help prevent and combat terrorism while also assisting governments in making informed decisions. Moreover, they contribute to enhancing citizens' awareness of the kind of terrorism activities a particular region is exposed to. In essence, these technologies provide cost-effective means to protect lives and the properties of the citizens [15] [16]. ML has the potential to predict terrorist acts by analyzing various data, including financial transactions, travel patterns, activities, and publicly accessible information from platforms like social media [17] [18]. The expected outcome of this study will highlight the importance of collecting global terrorism data and the ability to obtain useful information from them to counter its actions.

Studying and predicting terrorist attacks helps target these groups and gives important information for anti-terrorism efforts. This helps authorities find new or hidden terrorist actions/ plans quickly, reducing harm to people and property, preventing issues, and making society safer. Even though terrorist attacks may seem random, they are usually meticulously planned and deliberate. Attacks by the same groups or individuals often share certain characteristics, hence, there are patterns or informal rules guiding terrorist activities. By analyzing these patterns, authorities can make more detailed predictions and analyses, helping prevent attacks more accurately and giving more time for prediction. Researchers have applied ML methods to explore various aspects of terrorism [19]. Singh et al. at University of Connecticut [20] produced an adaptive safety analysis and monitoring (ASAM) system. He proposed the use of Hidden Markov models (HMMs) along with Bayesian Networks (BNs) to detect, predict and track potential terrorist activities in real time. This system was used in analyzing the vulnerabilities at the Athens 2004 Olympics. Another system built to counter terrorist online propaganda by trying to predict at what rate it is spread is discussed in [21]. They combined the GTD with 25,000 news sources from 1998 to 2016 considering different features such as social, natura, and geographical factors. This recommended system can help to predict and determine the risk for a future terrorist attack.

For the purpose of this study, we draw upon both the GTD as well as another dataset QFactors_Terrorism Dataset (QFTD) [22], and analyze them using ML techniques. Then we classify each terrorist group with the labels “Islamist”, “Left-wing”, “Separatist”, “Right-wing”, “Anti-colonial”, “Other” based on multiple sources. Given GTD, consists of 181.691 entries of terrorist attacks covering the period from 1970 up to 2017, we are able to explore overtime changes both on the density of those events and the repertoire of actions used by terrorist organizations. Python programming language was used to conduct the data cleaning and to analyze our data (via Google Colab). The dataset has records of both successful and failed attacks caused from a variety of terrorist groups. Various factors were taken into consideration, like the date of the attack, the terrorist group, where the attack took place, if there were any hostages, the number of victims, if there were any suicide attempts and how many days did the attack last. Moreover, the features provided by the QFTD, which are of economic and political nature were considered and their effect was investigated. For different sub-sets of each of the two mentioned datasets, a 5-layer DNN model is created and trained to predict the political label of the terrorist group, responsible for the attack [23] [24] [25]. The results from each dataset are compared to find the best one based on the feature correlations. In section Relevant Research, a literature review is provided, and the variety of methods and approaches is noted. In section Methodology, the datasets are first presented and described and the applied methodologies regarding the data pre-processing, feature selection, and the structure of the DNN model is described. Later, in the Results section the outcome of the models are presented and discussed and finally in the Conclusion section the findings of this thesis are summarized.

2. Relevant Research

The uptick of terrorist activities all over the world, is a very difficult matter that governments face. Utilizing Deep Learning, we can try to analyze and predict what kind of an attack we are dealing with and make better preparations to ensure the safety of the civilians. In the aftermath of the 11/9 terrorist attack and the more recent wave of terrorist attacks that shocked western societies, there was a need of a security system that could, first analyze the characteristics of those terrorist attacks and then, to predict the probability of future attacks. Machine learning and data mining can provide the essential tools to overcome and confront these type of future terrorist attacks. Acknowledging and understanding the potential factors leading to a terrorist attack was a tricky process. In 2017 [26] focused on the prediction of upcoming terrorist events from the GTD with data mining techniques. The GTD dataset contained over 140.000 events from 1970 - 2015 and 132 total features. the events are classified by the most common types of terrorist attacks: Political Terrorism, Separatist Terrorism, Religious Terrorism, Gangdom Terrorism. They combined two feature selection techniques Minimal-redundancy maximal-relevancy (mRMR) and Maximal relevance (Max-Relevance). Support Vector Machine (SVM), Naive Bayes (NB), and logistic regression (LR) were used on the selected features, and they demonstrated an accuracy of up to 78%. In [27] machine learning methods: DNN, SVM, and Random Forest, were used to evaluate the risk of terrorist attacks. The DNN with the highest area under curve value (0.96) consisted of 30 hidden layers, with sigmoid activation functions. They obtain 15970 sample data to train and test the performance of their models. Utilizing the “Boruta” package and the F-score metrics, they found that geographical, natural, and social factors are essential for simulation the risk of terrorist attacks in global regions. Additionally, their results suggests that also, the population density, nighttime lights, and the location (latitude, Longitude), play more important roles than the remaining factors in distinguishing whether an area belongs to a high-risk attack probability. Three-quarters of the sample data, which were almost evenly distributed, were randomly selected as training data, and the remaining were used as test data. With their models, they were able to predict the places where terrorist events might occur with 96% success rate.

In their 2019 study, S. Kalaiarasi [28] proposed an intelligent system based on machine learning to anticipate potential societal threats. They utilized the GTD and employed k-Nearest Neighbor (k-NN) and Random Forest (RF) algorithms for data classification. The k-NN algorithm demonstrated superior performance for the Weapon Classifier, while the Random Forest algorithm excelled for the Perpetrator Classifier. The GTD dataset was divided into an 8:2 ratio for model training and testing, respectively. The k-NN algorithm achieved an accuracy of 88.74%, while the RF algorithm achieved a precision of 90.45% and an accuracy of 89.95% for constructing the perpetrator classifier. A hybrid classifier was applied in a new framework for predicting terrorist attacks by X. Meng [29]. To optimize the weight of every classifier, a genetic algorithm was used. Out of the 113,114 records of the GTD, 45,221 were obtained, and the entries containing missing data were removed. Also, from the 134 attributes, they kept the seven that were most associated with terrorist attacks: city, specific area, type of attack, name of the terrorist group, type of weapon, damage caused, and produced ransom. For the feature analysis, they used discriminant analysis and concluded to four algorithms to classify their data, KNN, SVM, Decision Tree (DT) C4.5. The results showed that the hybrid classifier had better performance than the single classifier.

A hybrid DL platform based on Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM) models was proposed by Saidi and Trabelsi [30] to learn the temporal features and to predict future terrorist events around the world. An extension of the GTD containing around 190,000 events from 1970 to 2020. Regarding feature selection, the CNN was able to extract complex features and then, these features were provided to the LSTM model to learn the temporal correlation of the data. To prepare their dataset, the used LabelEncoder to convert text to numbers, SimpleImputer from sklearn was used to fill missing data and Synthetic Minority Over-sampling Algorithm (SMOTE) to deal with unbalanced classes. Five main features were used to train the model, if there was a case of self-inflicted suicide, success, weapon type, region, and the type of attack. The proposed CNN-LSTM model shows a correlation between the occurrence of weapon types in each attack and can accurately predict the success rate with 96% precision. For the multi-classification case, that of terrorist attacks, the improved DNN achieved an accuracy of 99%. Five different machine learning models, i.e., SVM, simple Neural Network (NN), Naive Bayes (NB), RF, and DT, were used to make predictions on attack type, attack region, and weapon type in 2018 by Verma et al. [31]. In order to find a significant relation between six features, the authors used rcorr() function in the Hmisc package which yields significant correlation for Pearson and Spearman correlations method. For this research, only the attacks from 2013-2016 contained 170,350 instances and 6 attributes. The response attributes are attacked type, attack region, weapon, the country, attack target, and success. The lit-wise deletion method is applied to handle the missing values in the dataset. The highest accuracy was achieved from RF for all predictions noting, attack type prediction 84%, continuing with 100% for region and 91% for weapon type.

Ze Li [32] analyzed and tried to predict how terrorist groups act, with a comprehensive framework that utilizes social network analysis, wavelet transformation, and pattern recognition approaches to understand the behavior of a terrorist group. Their framework was used to investigate Al-Qaeda data. They transformed their problem to behavior pattern recognition, having two classes, Normal and Attack behavior for a network. They managed to combine NN and Wavelet Natural Network with a three-layer structure to predict the group behavior. In 2020, a research article in Complexity [33], suggested a future terrorist activities prediction model, using Deep Neural Networks. They worked on the GTD containing 181,000 events from 1970 to 2018 and using 34 features, tried to predict different factors of a terrorist attack like if it there was a suicide, if the attack succeeded or not, the weapon types used, the region where the attack occurred, and the type of the attack. They followed standard data pre-processing techniques, where with the use of LabelEncoder of sklearn library they converted text to numbers, SimpleImputer to deal with missing data, SMOTE to deal with unbalanced classes and MinMaxScalar to normalize the data from -1 to 1. After 100 iterations, the single-layer (10 units) NN model showed lesser results, 72%, 80%, 78%, than the 5-layer (100-50-30-10-5 units) DNN model, which achieved 92%, 95%, 92%, for the weapon type, region, and attack type predictions respectively. Furthermore, they compared the previous NN models, with traditional ML algorithms like SVM, LR, and NB to show that the 5-layer DNN model outperformed all the others.

T. Jany [34] tried to predict the success of a terrorist attack. From the GTD 9 main features were selected: country, region, attack type, target type, weapon type, group name, suicide, multiple, and hostage. For data pre-processing, she performed one-hot label representation for each categorical feature, resulting to an input dimension of 3,512. Two different NN models were created using Adam optimizer, one 3-hidden-layer (100, 50, 20) units and trained for 50 epochs and one with four hidden layers of (50, 20, 10, 5) units. Both models had close prediction accuracy

on training set and on the dev set where the 3-layer model had 91.98% and 91.18% respectively. The 4-layer model had lower accuracy percentages, but the gap between train and dev was even smaller, 90.83% and 90.54%. A hazard grading model using Machine Learning techniques was published, to classify terrorist attacks using the Principal Component Analysis (PCA) method for features analysis with the help of K-Means Clustering [35]. The GTD contained of 180,000 terrorist attacks containing 45 features. Firstly, 14 indicators to the hazard of the attack we selected and then the PCA method, reduced the dimensions to 4. Then the k-means algorithm is used to cluster the terrorist attacks into 5 hazard levels according to the attack type. A pattern recognition approach was taken by [36] in the GTD which contained 27,000 events. They divided the provided features, creating new classes with their associated attributes. These are time (yea, month, etc.), location (country, region, city etc.), features (summary, multiple, succeed, suicide, et.), attack type (attack type 1-3), target (target type, nationality, etc.), weapon (weapon ty, weapon sub-type, weapon details, etc.) and terrorist group (group name, group motivation, etc.). They concluded that the Khi^2 formula is more effective than the Euclidean distance. Khi^2 formula, is used to reduce the size of a dataset, by evaluating similarities between the attributes. However, noted that experimental results with quantitative data should also be presented to test the effectiveness of the approach.

Other studies tried to analyze and predict what are the target of the terrorist attacks for the government to be more aware and enhancing their defenses. Hybrid classification algorithms were applied for terrorism prediction in Middle East and North America [37]. In this research it was studied, what groups where responsible for each attack in that region. Many traditional ML algorithms where tested, like NB, k-NN, DT, SVM, and RF. For their feature selection, they pointed out that the 10 most important ones were year, month, country, region, provstate, city, attack type, and target type. With their hybrid approach, RF was the most successful model, with a 99% accuracy. Another study, this time, regarding terrorist groups Iraq [38] was conducted to model their terrorist activity. Their GTD had events from 1970 to 2007 with 80,000 entries, but they used a subset of records, ranging from 1991 to 2007. Moreover, they used features like date, city, location of the incident, the type of weapons, the number of casualties, the number of wounded victims, the type of the attack, and the responsible terrorist group. K-means was used to cluster the data regarding the attack type using RapidMiner. Then, OneR classifier was applied and along with the use of WEKA data mining tool, they extracted the most important generated rules for the target feature. Finally, the Apriori algorithm, is applied to generate the association rules. It was found that there is a strong connection between the military target type, the explosives weapon type and the bombing attack type. In this independent study [39], the application of co-clustering method was discussed for pattern discovery. The author extracted textual data from a subset of GTD including events from 2010 to 2013. They found out a connection between time related features like months, geographical features like the country of the attack, with the terrorist groups.

In this research [40], the performance of classifiers such as Lazy Tree, Multilayer Perceptron, Multiclass and Naïve Bayes classifiers for observing the trends for terrorist attacks around the world were analyzed. The dataset included attacks from 1970 to 2015 reporting 156,772 events. The nine selected attributes for this research were month, year, region, weapon type, attack type, data source and property loss. Lastly, they created a new target feature, named attack responsibility, having three labels, “Claimed”, “Non-Claimed”, and, “Anonymous” based on if an organization has taken credit for the attack or not, or whether it is anonymous. Out of total reported attacks, 4,664 are claimed, 75,966 are non-claimed and 66,142 are anonymous. Olusola

et al. [41] developed an ensemble machine learning model to predict continents that might be prone to terrorism, using a SVM and KNN algorithms. For Feature selection, the used chi-squared, information gain and a combination of the two. As a result, 21 features were selected some of these are: month, day, region, location, if it was intentional, the level of violence threat, is suicide, attack type, target type, nationality of the target, group name, weapon type, number of casualties property damage and others. The results show that North America, South America, and Asia produced the highest Area Under Curve of 0.99. Subsequently, ensemble machine learning models were developed and employed utilizing the chosen features. Accuracy rates of 94.17%, 97.34%, and 97.81% were achieved with Chi-squared, Information Gain, and hybrid-based features respectively, in forecasting hazardous areas. Correspondingly, sensitivity scores stood at 82.3%, 88.7%, and 92.2%, while specificity scores were recorded at 98%, 90.5%, and 99.67% respectively. These findings indicate that the hybrid-based selected features yielded superior results compared to other feature selection techniques in predicting locations prone to terrorist activities.

In this thesis [22], they tried to create a classification model to identify the terrorist groups behind the attacks that have yet to be recognised. With the combination of the GTD and population-level demographic data from various open sources, a new dataset was released called QFTD. The used five different machine learning models Gaussian NB, Linear Discriminant Analysis (LDA), k-NN, DT, and RF and trained the separately on both GTD and QFTD dataset. Using RF classifier, a 68% accuracy was achieved on the new generated dataset. The utilization of Deep learning (DL) architectures and techniques for predictive modeling is applied in many different domains. DL is a subset of ML that utilizes hidden layers to reveal previously obscured spatial and temporal relationships in large datasets. In this thesis, we will analyze both GTD and QFTD and classify the events by the political terrorist group label based on different features [25] [23]. A 5-layer DNN model is going to be built and trained for each of the pre-processed sub-sets. Our models we are able to achieve overall more than 90% and used later to label the events whose responsible terrorist group political label is not set.

Table 1: The main results of the referred papers

Reference	Models	Best Accuracy	Feature Selection	Target
Terrorist Event Prediction Based on Revealing Data [26]	SVM, NB, LR	LR: 78.41%	Max-Relevance mRMR	Most Common Terrorist Type
Understanding the dynamics of terrorism events with multiple-discipline datasets and machine learning approach [27]	NNET, SVM, RF	RF: 96.6%	“Boruta” package F - Score	Global Risk of a Terrorist Attack

Using Global Terrorism Database (GTD) and Machine Learning Algorithms to Predict Terrorism and Threat [28]	k-NN, RF	Weapon k-NN: 88.74% Perpetrator RF: 90.45%	Mean Imputation	Weapons Used and Perpetrators
Big data-based prediction of terrorist attacks [29]	KNN, SVM, DT	DT: 90.5%	Correlation	Attack Type
A hybrid deep learning-based framework for future terrorist activities modeling and prediction [30]	DNN, CNN-LSTM	Weapon Type DNN: 99% Suicide CNN-LSTM: 99% Success CNN-LSTM: 94%	Correlation	Weapon Type, Suicide, Success
Predictive Modeling of Terrorist Attacks Using Machine Learning [31]	SVM, ANN, NB, RF	Attack Type RF: 84% Region RF: 100% Weapon Type RF: 91%	Linear Regression	Attack Type, Region, Weapon Type
Prediction of Future Terrorist Activities Using Deep Neural Networks [33]	Single-layer NN, 5-layer DNN, LR, SVM, NB	Weapon Type DNN: 92% Region DNN: 95% Attack Type DNN: 92%	Correlation	Success, Suicide, Weapon Type, Attack Type, Targeted Region
Predicting Success of global terrorist activities [34]	3-layer DNN, 4-layer DNN	3-layer DNN: 91.98%	Correlation	Success
Hybrid Classification	NB, k-NN,	RF: 99%	Correlation	Terrorist Group responsible for

Algorithms for Terrorism Prediction in Middle East and North Africa [37]	DT, SVM, RF			attacks in Middles East and North Africa
An ensemble machine learning model for the prediction of danger zones [41]	SVM combined with KNN	On Hybrid feature selection: 97.81%	Chi-Squared, Information Gain, Hybrid	Area of terrorist attack
An Integrated Machine Learning Approach To Studying Terrorism [22]	RF, K-NN, DT, NB	RF: 68%	Correlation	Terrorist Group

3. Methodology

3.1 Study Flow

In this study, standard workflow was followed (see Figure 1), that includes preprocessing, feature selection, training, testing and prediction. We analyzed two different datasets to understand their given features and evaluate their suitability for the prediction tasks. One was the GTD maintained by researchers at the National Consortium for the Study of Terrorism and Responses to Terrorism (START) and provided by Kaggle (see link). The other one was the QFactors Dataset, released from the thesis of Andi Peng "An Integrated Machine Learning Approach to Studying Terrorism" [22]. We will analyze each dataset more thoroughly in the relevant section. To further enhance the datasets, we created the new political label column and then they were pre-processed to remove duplicates, fill missing data, deal with text values, normalize the data and select our key features. To fill in the missing data, we used different techniques. To fill the missing numerical data, we used the mean value, and for the textual data, we filled them with the "Uknown" value, as it already exists in our features. To select the most important features, we capitalized different ML algorithms, like DT, RF and LDA. In the end, the pre-processed data was split into training, validation, and test with the rates of 5:3:2, 6:2.5:1.5 and 7:2:1 respectively. Then we created different Deep Learning models, to compare their performance and select the most fit to classify the terrorist events by the political identity of the terrorist actors. The DNN models contained 1 input layer, 4 hidden layers (100, 50, 30, 10 units respectively) and one output layer of 6 units, as the number of classes that we want to predict. For the output layer, the softmax activation function was used, along with categorical cross entropy utilizing Adam optimizer. The models were trained from the training set and then evaluated on the test set. In the end, we tried to classify the "Unknown" values of each of our target labels. These components are described in the following sub-sections.

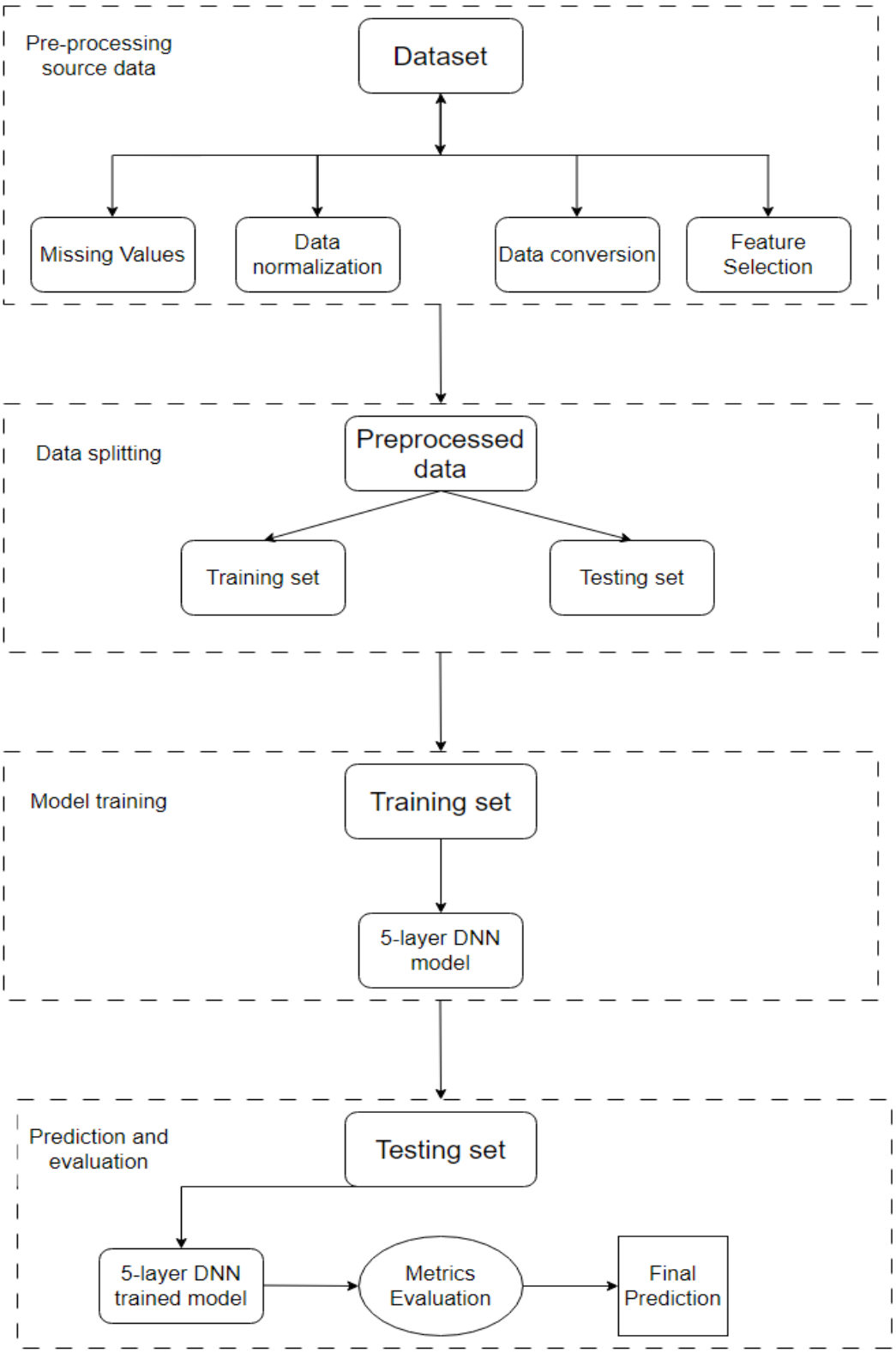


Figure 1: Methodology Steps

3.2 Data description

In this section the two datasets will be analyzed and explained. The provided features will be examined to have a better understanding of the problem and evaluate their relevance. It is important to mention that for an incident to be categorized as a terrorist attack, it should meet at least two of these three conditions: The events that took place were intentional, acts of violence were occurred, if the perpetrators responsible for these violent acts were a sub-national group.

3.2.1 Global Terrorism Database

The Global Terrorism Database (GTD) is an open-source database including information on terrorist attacks around the world from 1970 through 2017. The GTD includes systematic data on domestic as well as international terrorist incidents that have occurred during this period and now includes more than 180,000 attacks. The database is maintained by researchers at (START), headquartered at the University of Maryland. The entries of this dataset are a collection of various incidents from global news sources [42]. The GTD contains 181,691 instances of recorded incidents of terrorism attacks recorded from July 1970 to December 2017 from several countries around the world. Out of these attacks, 161,632 succeeded and were labeled with 1, and the rest 20,059 attacks failed and labeled with 0. The dataset has 134 attributes, many of which are ignored due to irrelevance or due to a large amount (85%) of missing data. There is also detailed description of this dataset in [42] [43] [44]. In Figure 2, we can see the growth of the terrorist attacks through the years. Many historical events happened during this timeline that were greatly affected. To have a better understanding of our data, it is very helpful to visualize the number of attacks through the years.

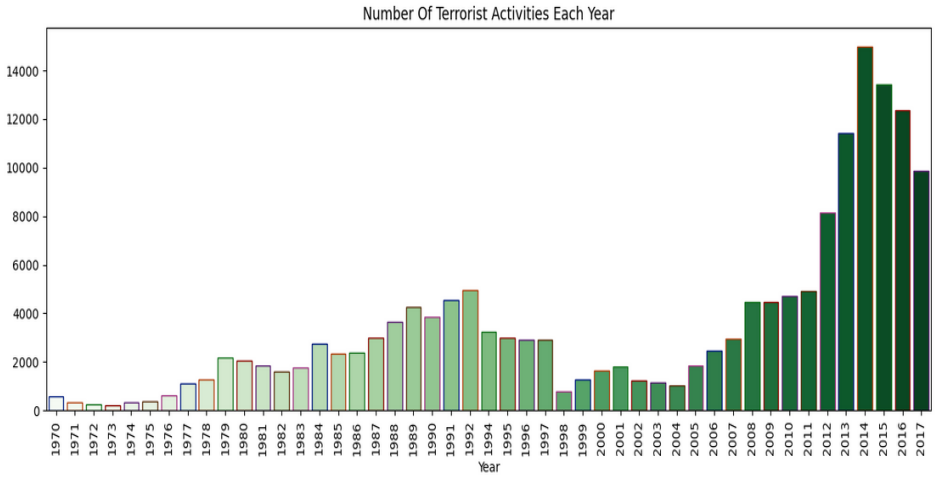


Figure 2: Terrorist Activity through the years

We can see that during the last decade, the number of terrorist attacks, has been significantly increased. Social scientists have identified various reasons for the increase, such as the rise of Islamist terrorist movements, ethno-cultural conflicts in Africa, and the increase of the right-wing terrorism in Nordic countries [45] [46] [47]. During these years, the Region with the highest recorded terrorist attacks, is the Middle East & North Africa (50,474). In 2000, North America had a lot of casualties, but we observe that during 2015, Middle East & North Africa had the most recorded entries in the database. Iraq (24,636), Pakistan (14,368) and Afghanistan (12,731) are the top three most attacked countries. In the map in Figure 3, we can see the attacks per region. We will see that it is indeed a global phenomenon. The most casualties were recorded in the United States during 9/11, where in the dataset it is recorded that 1,570 people were injured or killed. In Figure 4 the number of casualties per region are displayed through the years.

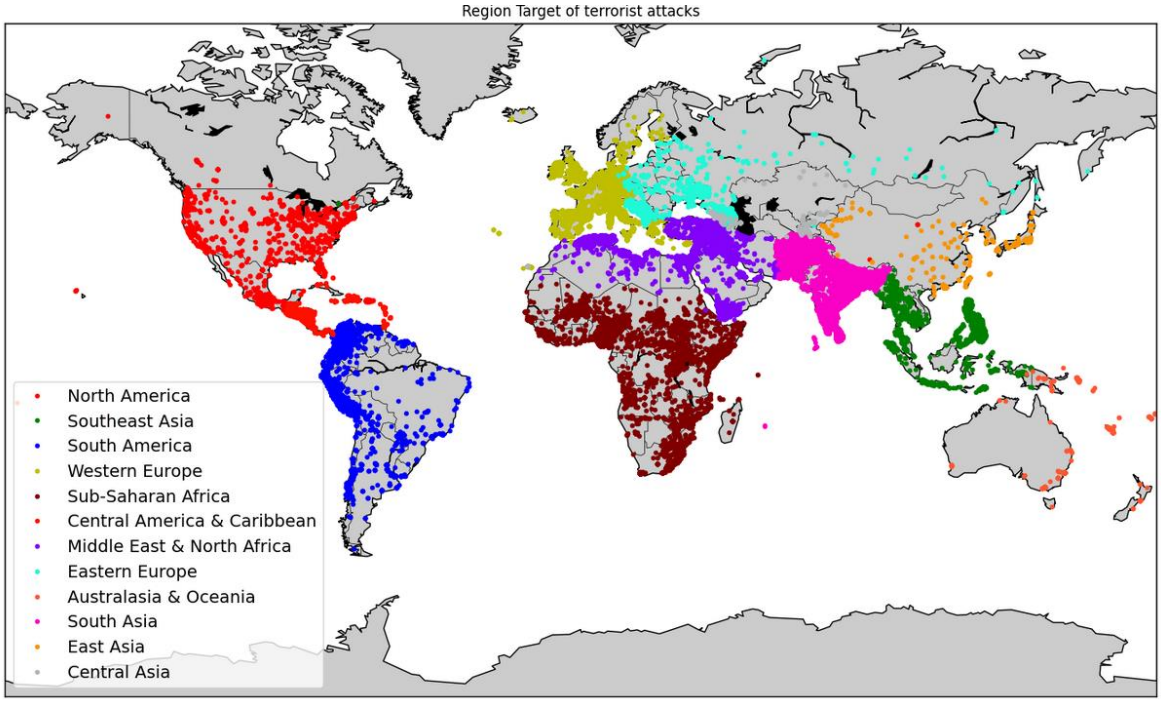


Figure 3: Terrorist Attacks Per Region

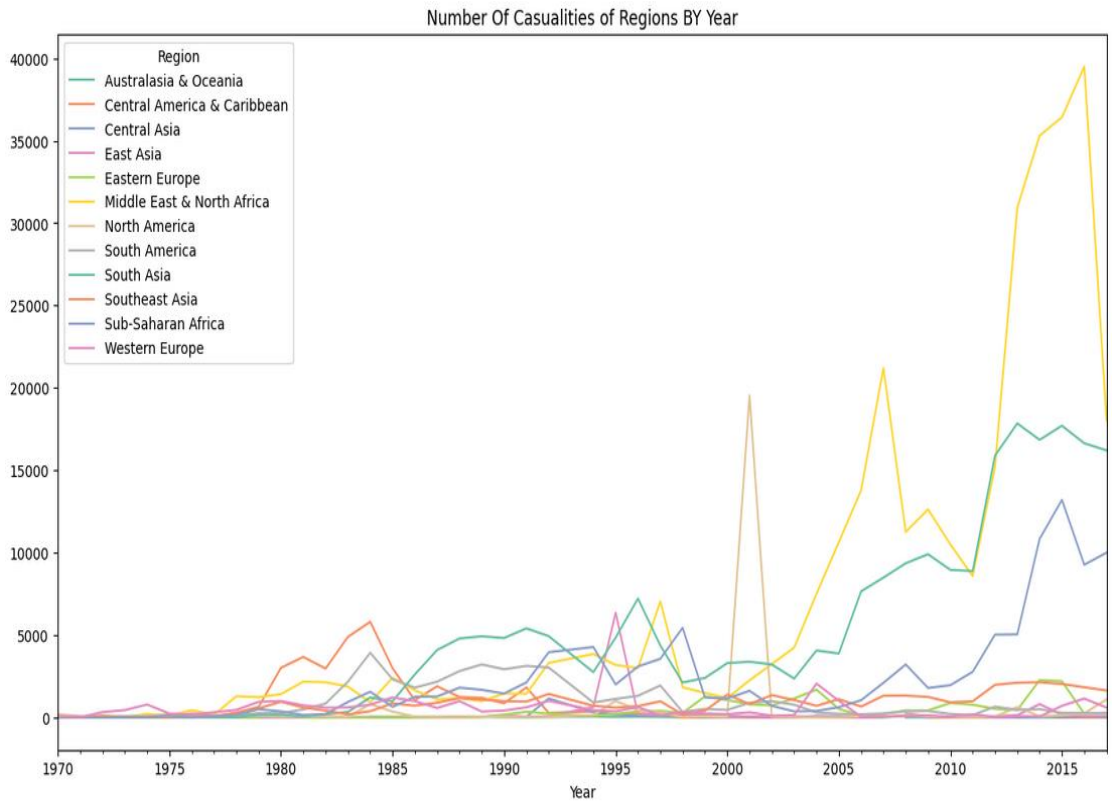


Figure 4: Region casualties through the years.

It is also very interesting to see some of the groups with the most terrorist activity (Figure 5). These terrorist groups are linked with geographic location of an attack. So, it is legitimate that the most active terrorist groups are from the Middle East & North Africa region, because, as we said previously, perpetrators responsible for these attacks tend to be a sub-national group. We can see though, that for the most terrorist activities, we haven't identified the responsible group. The most active terrorist actors seem to have jihadist background, while left-wing organizations follow with a significant number of terrorist attacks. While it is quite difficult to identify the ideological characteristics of those groups, this study attempts to provide a broad categorization of those perpetrators. Therefore, for the purpose of this study, we separated those groups into five board (and not exclusive) categories, namely, "Islamist" groups, "Left-wing", "Right-wing", "Separatist" actors, and "Other".

Unknown	77351
Taliban	6800
Islamic State of Iraq and the Levant (ISIL)	4723
Shining Path (SL)	3992
New People's Army (NPA)	2651
Farabundo Marti National Liberation Front (FMLN)	2530
Al-Shabaab	2386
Revolutionary Armed Forces of Colombia (FARC)	2304
Kurdistan Workers' Party (PKK)	2254
Basque Fatherland and Freedom (ETA)	1849
Communist Party of India - Maoist (CPI-Maoist)	1824
Boko Haram	1753
Irish Republican Army (IRA)	1679
Maoists	1545
Liberation Tigers of Tamil Eelam (LTTE)	1519
National Liberation Army of Colombia (ELN)	1481
Tehrik-i-Taliban Pakistan (TTP)	1303
Palestinians	947
Al-Qaida in the Arabian Peninsula (AQAP)	917
Houthi extremists (Ansar Allah)	826

Figure 5: Number of terrorist attacks per Group.

In Figure 6 and Figure 7 we will see how these attacks are spread across the world through the years. In Figure 6, the region each terrorist group tends to act can be observed. Apart from the incidents that took places in Europe [48], there are plenty of terrorist attacks in different geographical region. Most of those attacks took place in Latin America, Southern Africa, Middle East, and Asian countries. Explanations for these tendencies rely on the case-specific characteristics of these regions. In Latin America countries, groups related to drug trafficking and left-wing paramilitary organizations are the main penetrators of terrorist attacks [49]. In Africa, terrorist attacks can be understood as a part of the internal ethno- cultural conflict which on some occasions may lead also to civil wars [50]. Lastly, in Asian countries the main perpetrators are jihadist terrorist organizations which have managed to establish their cells in different countries and regions. Figure 7 shows, how active each group has been through the years. Very interesting is the uptick of terrorist attacks after 2000. As Figure 7 suggests, the new wave of jihadist terror attacks can be seen as the main explanation for this sharp increase at the level of terrorism [51].

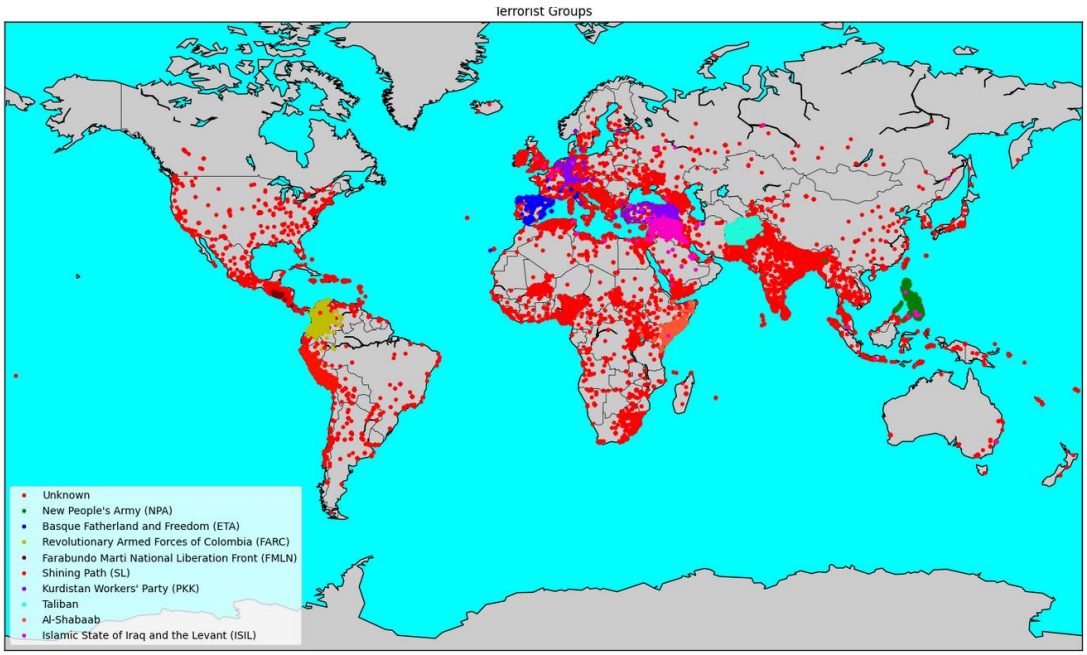


Figure 6: Terrorist activity across the world.

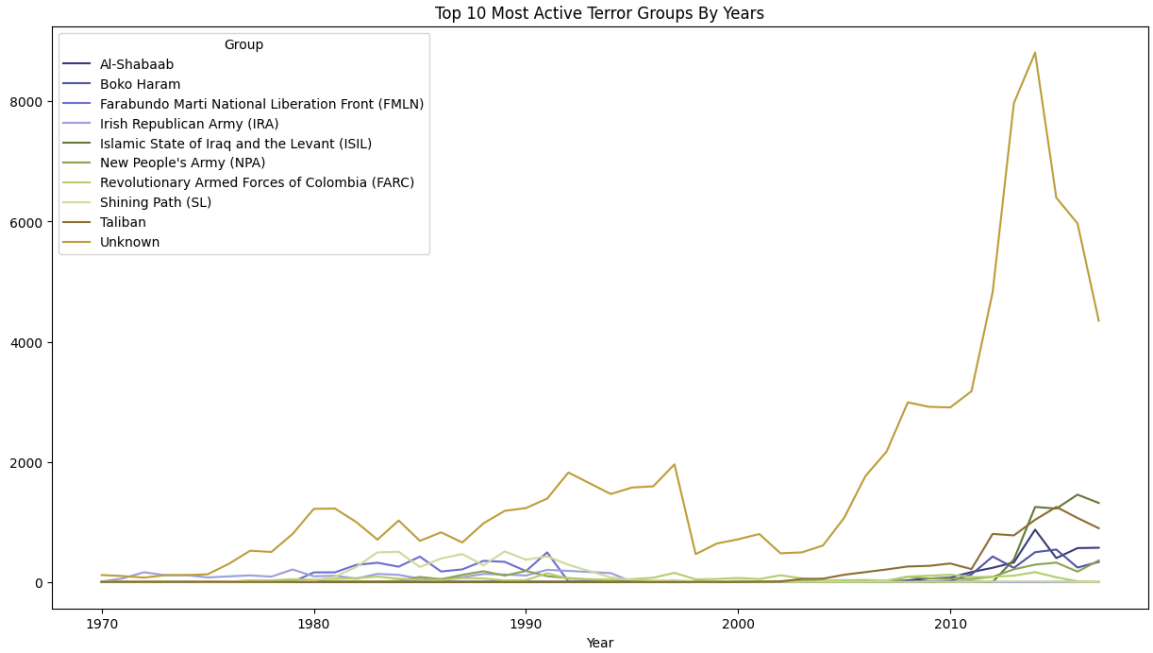


Figure 7: Terrorist activity through the years.

The most common attack methods are bombing or via explosions with a big difference from the others (Armed Assault, Assassination, Infrastructure Attack, Hostage/Kidnapping, unarmed assaults, hijacking) Figure 8. This information is very critical for the outcome of a terrorist attack and it's a very important factor for our study. We can also see here that for some attacks we don't

know what weapons and methods were used. Out of the 181,691 entries, the 15,157 are labeled as “Uknown” weapon types which is the 8.34% of our total dataset. This is something we will try to do with our prediction models and will explain in the related section.

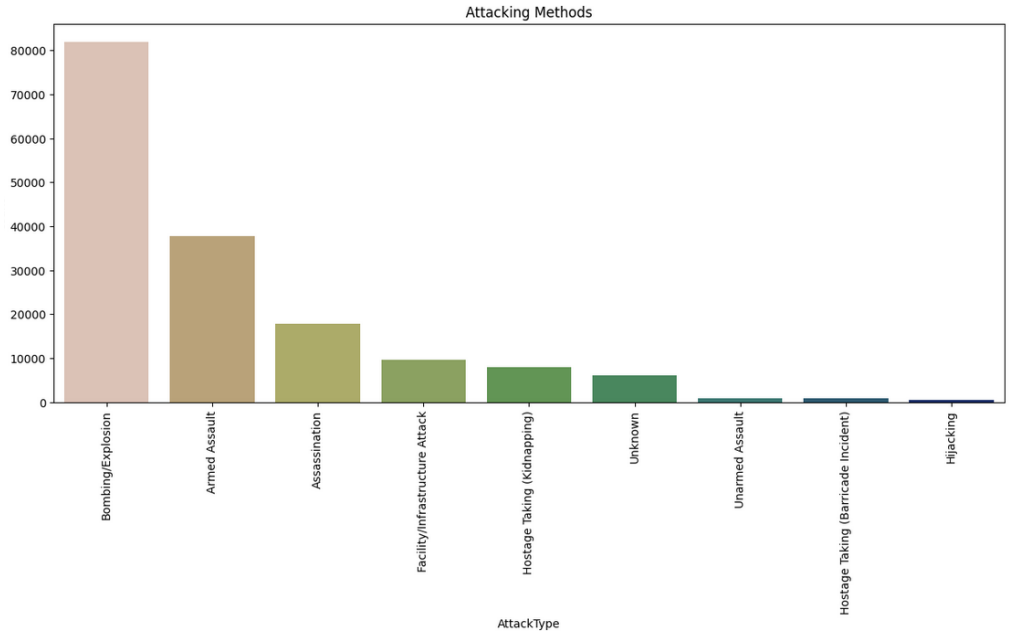


Figure 8: Terrorist Attack Methods GTD

In Table 2, we explain all the attack types provided to understand better how each entry was labeled.

Table 2: Attack Type Feature Description

ATTACKTYPE1	ATTACKTYPE1_TXT	Description
1	Assassination	This act is primarily intended to cause the death of one or more specific and notable individuals, typically those holding significant positions such as high-ranking military officers, government officials, or celebrities. It specifically targets individuals of prominence and does not involve attacks on members of a non-specific group. For instance, the killing of a police officer would be categorized as an armed robbery unless there is evidence suggesting that the assailants specifically aimed to assassinate a particularly distinguished officer
2	Armed assault	An assault primarily directed at causing direct physical harm or fatality to humans, achieved through the utilization of a firearm, incendiary device, or sharp

		instrument (such as a knife). This category excludes attacks involving the use of fists, stones, sticks, or other (less lethal) handheld weapons.
3	Bombing/explosion	An assault characterized by the primary effects generated from a material that rapidly decomposes, creating a pressure wave leading to physical damage in the surrounding environment. This may involve high or low explosives, including a dirty bomb, but excludes a nuclear explosive device releasing fission and/or fusion energy or an incendiary device with a much slower decomposition rate. If an attack combines specific explosive devices with firearms, incendiary tools, or sharp objects, it is classified solely as an armed assault. The subcategories of explosive devices falling under this classification include grenades, projectiles, and unspecified or other explosives.
4	Hijacking	An intentional act aimed at seizing control of a vehicle, such as an airplane, ship, bus, etc., with the intention of diverting it to an unplanned destination, securing the release of prisoners, or achieving some political goal. While obtaining a ransom payment should not be the sole motive behind a kidnapping, it may be one aspect of the incident, provided that other objectives have been declared. Kidnappings are differentiated from hostage-taking as the primary focus is on the vehicle, irrespective of whether there are individuals or passengers inside.
5	Hostage taking (barricade incident)	An action primarily carried out to attain a political goal by gaining control of hostages through concessions or by disrupting regular operations. These attacks are set apart from kidnappings as they typically occur at the target's location and usually do not involve the intention of holding the hostages for an extended period in a separate underground location.
6	Hostage taking (kidnapping)	An action carried out with the aim of seizing hostages to achieve a political objective through concessions or the disruption of regular activities. Kidnappings are distinct from barricade incidents (as described above) since they entail the movement and retention of hostages in a different location.
7	Facility / infrastructure attack	An action, without the use of explosives, primarily intended to cause harm to a non-human target, such as a building, monument, train, oil pipeline, etc. These attacks may include arson and various types of sabotage (e.g., sabotaging a railway is considered an attack on infrastructure, even if it results in harm to passengers). Facility or infrastructure attacks may

		involve actions intended to harm a facility but may also incidentally affect the surrounding people (e.g., an arson attack aimed at damaging a building that causes injury or death in the process).
8	Unarmed assault	An assault primarily intended to cause direct physical harm or death to humans, excluding the use of explosives, firearms, incendiaries, or sharp instruments (e.g., knives). This is because attacks involving chemical, biological, or radiological weapons are categorized as unarmed assaults.
9	Unknown	The attack type has not been determined from the information available.

The most prominent target types seem to be private citizens and property, military bases, police, government, and businesses as seen in Figure 9. Private property attacks have greatly increased from 2010 and after. All these features that we mentioned and analyzed are key factors to our research as they also and the factors that we will try to predict. We experimented using various lists of features for our training dataset, to compare the overall performance of our models. In the next section, we will explain how the features of the second dataset differs from this one. Here the unknown attack targets are 7,276 which is 4.01% of our dataset.

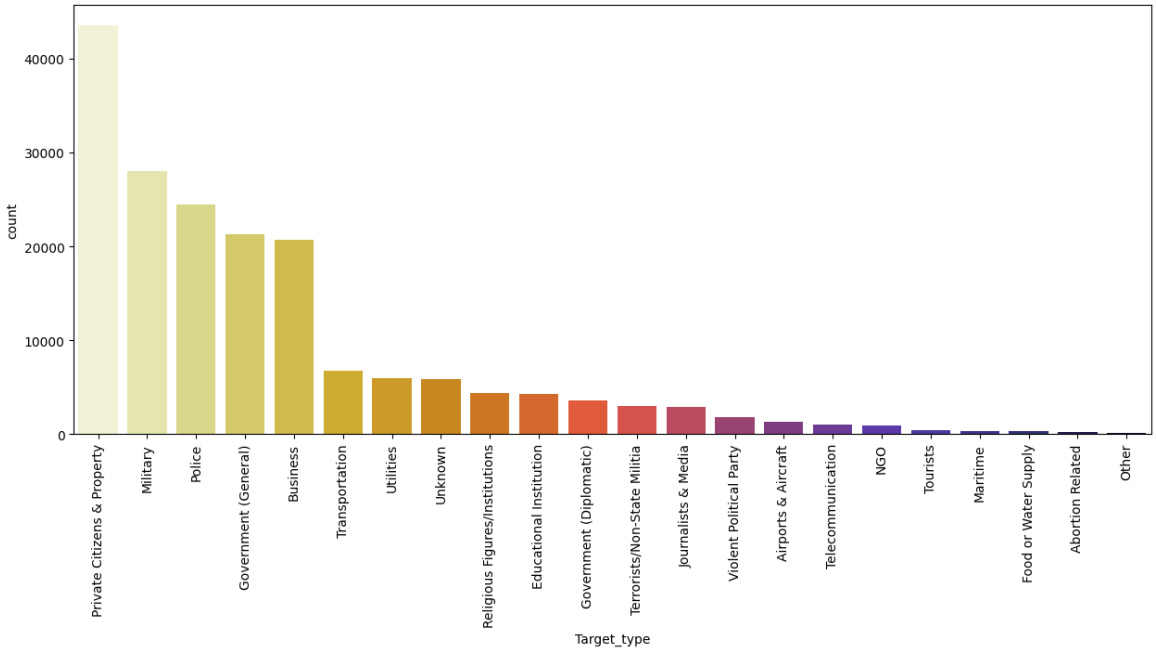


Figure 9: Terrorist Attack Targets GTD

In Figure 10, we analyze what weapons were used in these terrorist events. It very clear, that most of the times, terrorists use explosives and firearms that cause massive damage to the public and threat great geographical radius. Having in mind most of the world shacking terrorist attacks, we can potentially connect the usage of explosives with the action of jihadist groups. Governments must be very careful on how to deal with such dangerous means. Knowledge on what means might be used during a terrorist attack can save many lives if the correct counter measures are taken. Additionally special forces in each country could pay more attention not only to the potential perpetrators of these terrorist attacks but also to the channels thought which those terrorists managed to supply those explosives.

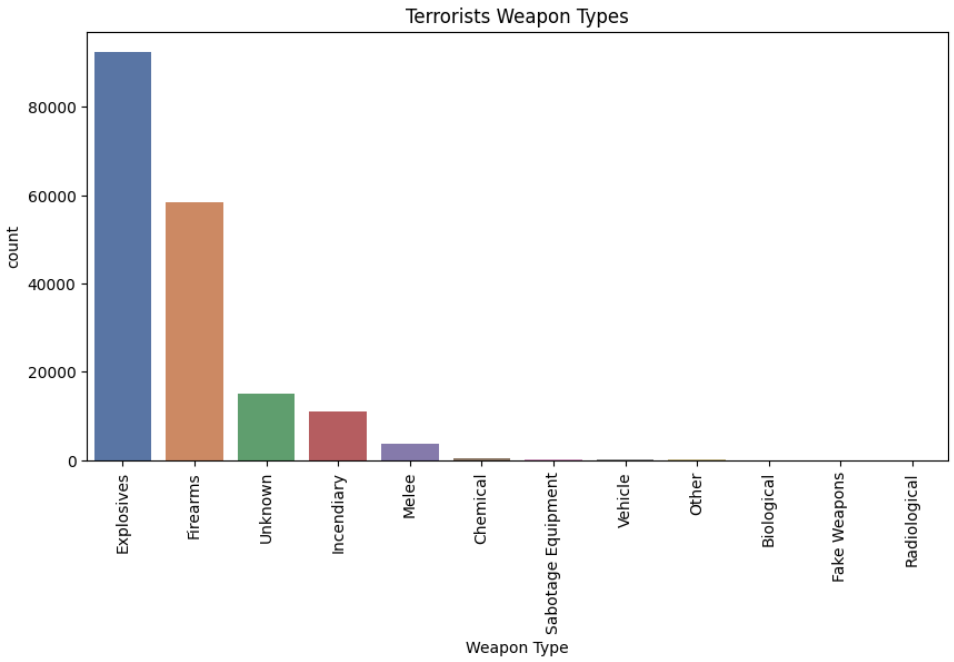


Figure 10: Terrorist Weapon Types GTD

3.2.2 QFactors Dataset

The Qfactors dataset, provided by Andi Peng's thesis [22], introduces some new features to study a more population-level demographic approach. It is a result of the GTD dataset that we mentioned previously, World Bank, United Nations, and other open date sources. These new features are explained in the bellow table (Table 3). This dataset consists of only 60,095 records with 54,454 being a success and 5,642 fail from 1990 to 2016. Unlike the GTD, it contains incidents from 107 different countries (GTD has 205 countries) and 2,057 different terrorist groups (GTD has 3,537). Finally, both datasets share the same amount of attack types.

Table 3: QFDT Features Description

Feature Information	Dataset	Source
education level	Human Development Report (HDR)	United Nations Development Programme (UNDP)
number of immigrants number of refugees	International Migration Stock	United Nations Department of Economic and Social Affairs (UN DESA)
GNI per capita (PPP) life expectancy population density primary school enrollment	World Development Indicators (WDI)	The World Bank
violence by government	Armed Conflict Dataset	Uppsala Conflict Data Program (UCDP) / Peace Research Institute Oslo (PRIO)
political freedom	Human Freedom Index (HFI)	Cato Institute

We will also, like the GTD database, analyze the target values that we are going to classify. For the attack type feature, we have 1,902 unknown values out of the 60,096 in total which is the 3.17% of our dataset (Figure 11).

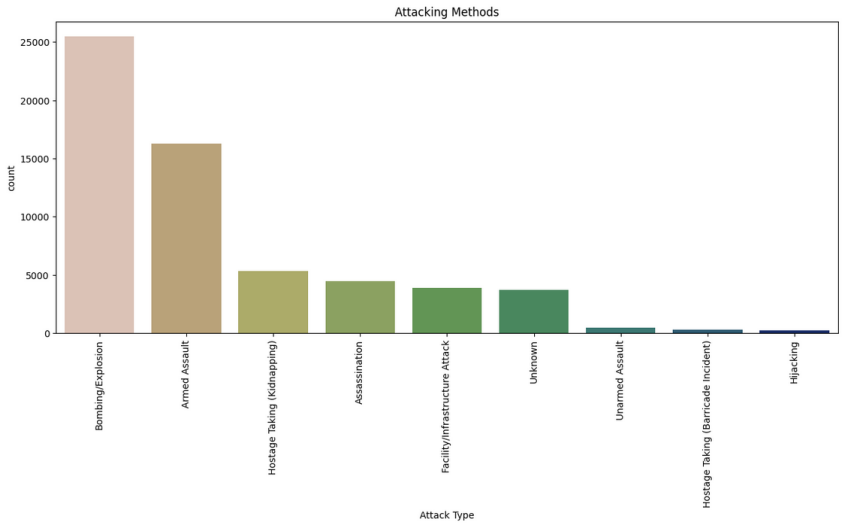


Figure 11: Terrorist Attack Types QFactors

As we can see, for both datasets the main type of attack is via Bombing/ Explosions and Armed Assault (Figure 12). So, we can see that there are indeed some preferred methods. For the weapon type feature we find similar results. We got 6484 (10.78%) values labeled as unknown. Again, we see that the most common weapons used are explosives and firearms.

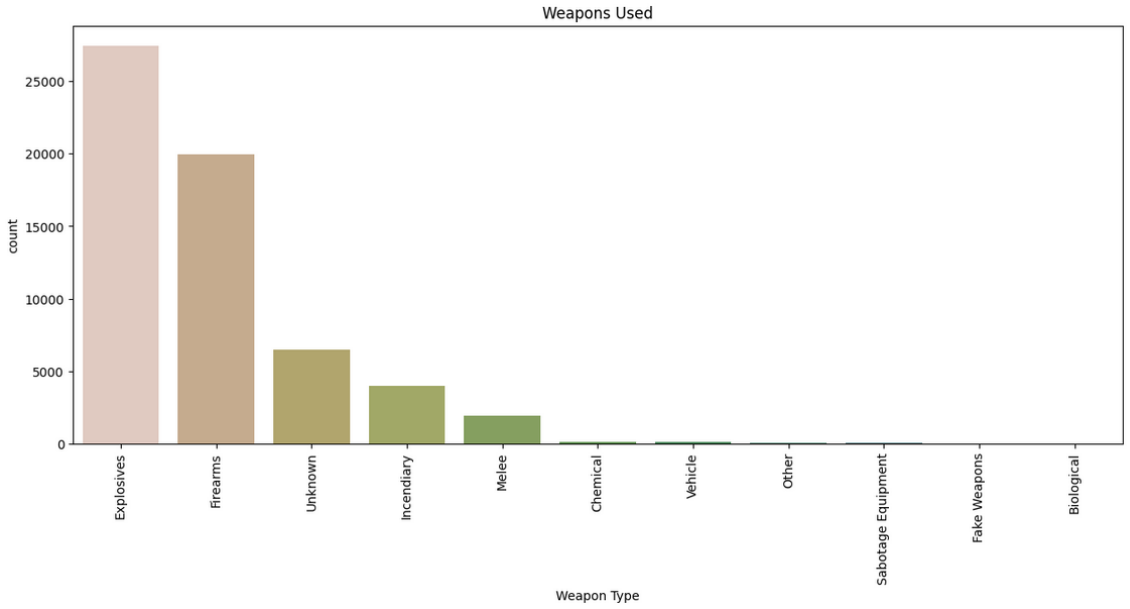


Figure 12: Weapon Types QFactors

In the Qfactors dataset we have 1293 unknown values regarding the Attack Targets, which is 2.15% of our data. In Figure 13, we will see that both the GTD and QFactors dataset, share the same distribution for the attack targets. In this thesis, after analyzing and pre – processing these two datasets, we proceed in creating our DNN models. Each model was trained on either the GTD or the QFactors dataset. The most capable models were selected at the end and used to try and classify the unknown values for our target label. Finally, after applying the PCA method in both datasets, we trained our PCA DNN models and used them to predict the values of each dataset.

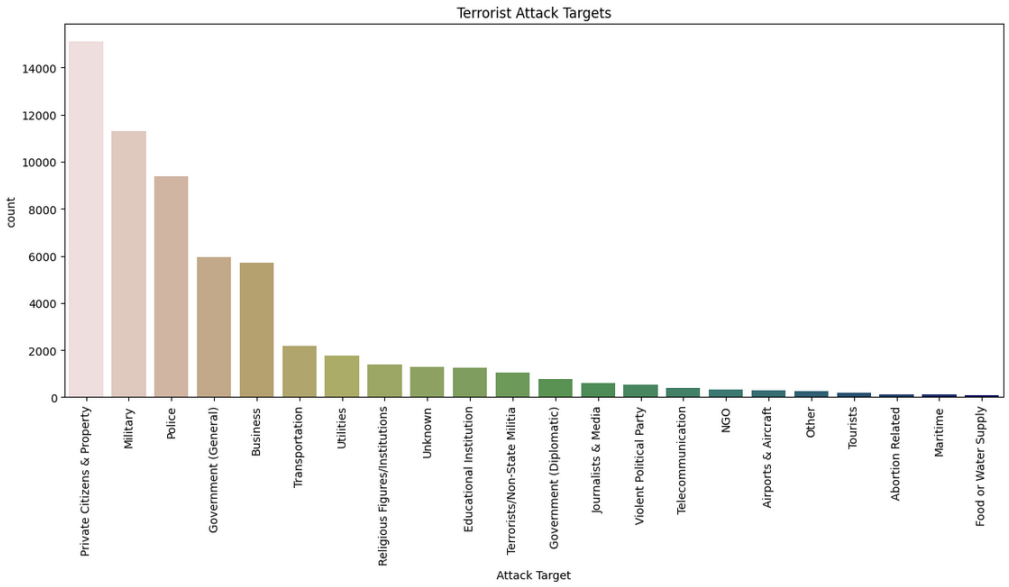


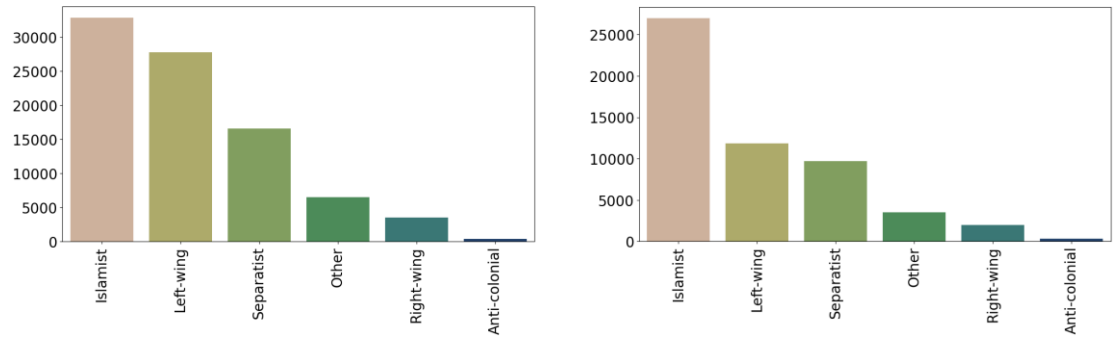
Figure 13: Terrorist Attacks Targets QFactors

3.2.3 Political Label Feature

In this thesis, depending on their political orientation, the terrorist groups are divided into six new groups: “Islamist”, “Left-wing”, “Right-wing”, “Separatist”, “Anti-colonial”, and “Other”. This new label is also the target feature of this thesis’ classification problem. There are similarities between the events caused by each group, but there are also some differences that need to be pointed out. The two datasets include a different variety of terrorist groups, as they are constructed from different timelines, the GTD is from 1970 to 2017 and QFTD is from 1990 to 2016.

Figure 14, displays how the terrorist groups are divided in each dataset. Difference can be seen between the occurrence of Left-wing attacks, where there are lesser in the QFactors dataset. In Figure 15, it can be seen, using the GTD data, that the geographical region differs. Each group tends to be more active in certain areas. Islamist groups is more active in Middle East & North Africa region (13,898 attacks), following with South Asia (10,373 attacks) and Sub-Saharan Africa

(6,291) with attacks recorded in Afghanistan, Iraq, Somalia, Nigeria and more. On the other hand, the Left-wing group, has more recorded attacks on South America (11,943), Central America & Caribbean (5,307) targeting countries like Colombia (5,180), Peru (5,106) and El Salvador (3,672). The most attacked region by the Right-wing group, is Western-Europe (1,070) where the 716 attacks were recorded in United Kingdom. Separatists are more active in Western Europe (6,040) and Middle East & North Africa (4,228) with the most ones in United Kingdom (2,691), Turkey (2,151), Spain (2,060) and Sri Lanka (1,269). Anti-colonials have attacks recorded only in Sub-Saharan Africa region with most being in Angola (394). Like the GTD, is the distribution of the events, in the QFactors dataset, even though the records are from 1990 till 2016.



(a) GTD (b) QFTD

Figure 14: Terrorist groups divided by political label.

In matter of the terrorist activity throughout the years, from

Figure *16* shows that Islamist have become a lot more active since 2010. The rest of the groups show a lot of activity from 1980-1990, which rises again after 2010. All groups share similar attacking methods, with the most common being bombing, armed assaults and assassinations. Military and private citizens' properties are the most common attack targets. Also, Islamist and Right-wing seem to have more attacks on religious institutions. The left-wing has cause more attacks on journalists and media.

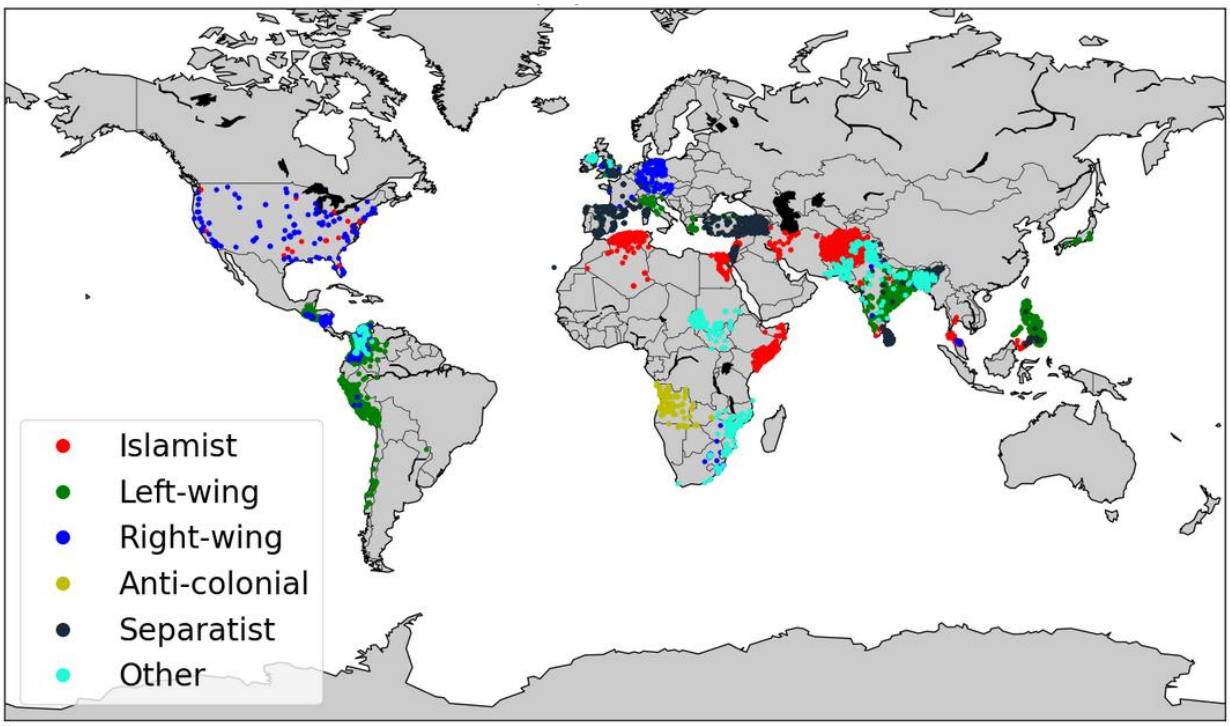
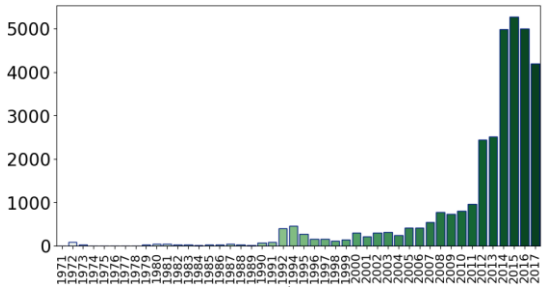
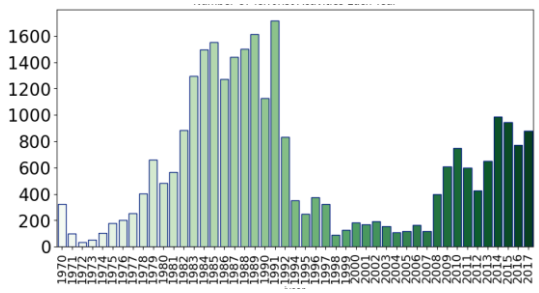


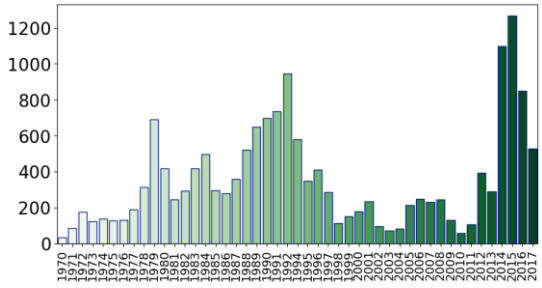
Figure 15: Terrorist Activity Map by Political Label



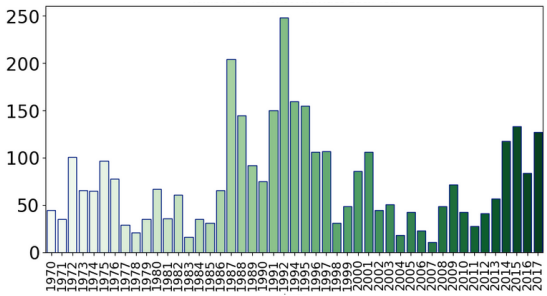
(a) Islamist



(b) Left-Wing



(c) Separatist



(d) Right-Wing

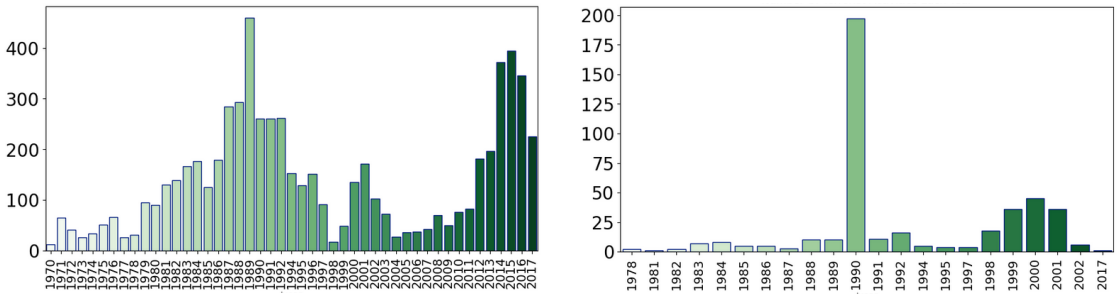


Figure 16: Terrorist Group activity timeline based on the political label.

3.3 Feature Selection and Data pre-processing

During this research, a variety of different feature sets were used to train the prediction and classification models. The steps that we followed on the selected features to pre-process them were data cleaning, text to numbers and data normalization. These steps are further analyzed in this subsection. The data modifications were made with the use of Python programming language inside the Google Colab environment. Moreover, we applied various ML algorithms to see the importance of our feature and try to create a sub-set of 8 features that will contain the most information available.

The Decision Tree Classifier was the first one we investigated. Decision Tree is a supervised learning algorithm used for both classification and regression tasks in machine learning [52] [53]. It operates by partitioning the feature space into a set of hierarchical decision rules based on the values of input features. Decision Tree selects the best feature to split the dataset at each node based on a chosen criterion, such as Gini impurity or information gain (entropy). Then the selected feature is used to split the dataset into subsets (or child nodes) based on the feature's values. Each subset contains data points that satisfy the condition of the split. The process of feature selection and splitting is recursively applied to each subset (child node) until one of the stopping criteria is met. Stopping criteria may include reaching a maximum tree depth, having a minimum number of samples in a node, or achieving perfect purity (homogeneity) in the node. Once a stopping criterion is met, a leaf node is created, which represents the final prediction for the data points that reach that node. In classification tasks, the prediction for the leaf node is determined from the most occurred class. For regression tasks, the mean or median of the target values in the leaf node is assigned as the predicted value.

The second technique we tried was Random Forest. RF is an ensemble learning method used for both classification and regression tasks in machine learning [54] [55]. RF operates by creating numerous decision trees during training and outputs the mode of the classes (classification) or the mean prediction (regression) from the individual trees. RF begins by randomly selecting n samples from the training dataset with replacement (bootstrap sampling). This creates multiple subsets of the original dataset; each called a bootstrap sample and for each sample, a decision

tree is constructed. However, Random Forest introduces randomness during the construction of each tree. For each node of the tree, a random subset of features is selected from the full set of features. This subset is typically much smaller than the total number of features. The tree is grown to its maximum depth without pruning, resulting in a deep but less correlated ensemble of decision trees. Once all trees are constructed, predictions are made for each tree. For classification tasks, each tree votes for the most popular class among the observations in the terminal node that the observation falls into. For regression tasks, each tree's prediction is averaged to obtain the final prediction. For classification tasks, the class that receives the most votes among all the trees is chosen as the final for the prediction. For regression tasks, the average of all tree predictions is taken as the final prediction. The randomness introduced during the construction of each tree and the aggregation of predictions from multiple trees help reduce overfitting and improve generalization performance.

Finally, we applied Linear Discriminant Analysis. LDA is a dimensionality reduction technique and a classification algorithm used in machine learning and statistics [56]. It aims to find a lower-dimensional space that maximizes the separability between different classes in the data. In LDA, the goal is to find a set of linear combinations of the input features (independent variables) that best separate the classes (dependent variable). This is achieved by maximizing the between-class scatter while minimizing the within-class scatter. LDA calculates the scatter matrix for each class, which represents the spread of data within each class. Mathematically, SW is calculated as the sum of covariance matrixes of each class weighted by their respective class probabilities.

$$SW = \sum_{i=1}^{c} p_i \, \Sigma_i$$

Where c is the number of classes, p_i is the probability of class i, and Σ_i is the covariance matrixes of class i. LDA calculates the scatter matrix between classes, which represents the spread of data between classes. Mathematically, SB is calculated as the sum of the covariance matrix of the class means weighted by their respective class probabilities.

$$SB = \sum_{i=1}^{c} p_i(\mu_i - \mu)(\mu_i - \mu)^T$$

Where μ_i is the mean vector of class i, μ is the overall mean vector, and p_i is the probability of class i. LDA finds the linear combinations of features (eigenvectors) that maximize the ratio of between-class scatter to within-class scatter. These linear combinations are the new axes in the reduced-dimensional space.

$$SW^{-1}SB$$

Finally, LDA projects the original data onto the new subspace formed by the selected eigenvectors. LDA is commonly used for dimensionality reduction and classification tasks, particularly when the classes are well-separated, and the assumptions of normality and equal covariance matrices hold. It provides a way to reduce the dimensionality of the feature space while preserving the class-

discriminatory information, making it useful for visualization, feature extraction, and classification purposes.

3.3.1 Global Terrorism Dataset (START)

The first step taken was data cleaning. This process is very important and affects greatly the training process of our models. One of the biggest challenges is dealing with missing data. For our study, we set a threshold of 85% and dropped any columns that had more than 15% missing data. With this way, our dataset was reduced to 66 features. Let's explain these remaining features in Table 4 and then decide which we are going to use for our research.

```
threshold = 0.85  # 85%

# Calculate the percentage of null values for each column
null_percentages = data.isnull().mean()

# Identify columns with more than 85% null values
columns_with_high_nulls = null_percentages[null_percentages > threshold].index
columns_with_high_nulls = pd.Index(columns_with_high_nulls.tolist())

data = data.drop(columns=columns_with_high_nulls)
```

Table 4: GTD Features explained.

Feature	Description
eventid	Unique identifier for each terrorist event.
iyear	Year of the incident.
imonth	Month of the incident.
iday	Day of the incident.
extended	Indicates whether the incident extended beyond 24 hours.
country	Numeric code for the country where the incident occurred.
country_txt	Name of the country where the incident occurred.
region	Numeric code for the region where the incident occurred.
region_txt	Name of the region where the incident occurred.
provstate	Province or state where the incident occurred.
city	City or location where the incident occurred.
latitude	Geographic coordinates of the incident location.
longitude	Geographic coordinates of the incident location.
specificity	Describes the geographical specificity of the incident location.
vicinity	Indicates whether the incident occurred in the immediate vicinity of the city.

location	Detailed location of the incident.
summary	Brief summary or description of the incident.
crit1	First criterion indicating the incident meets specific conditions.
crit2	Second criterion indicating the incident meets specific conditions.
crit3	Third criterion indicating the incident meets specific conditions.
doubtterr	Indicates doubts about the veracity of the incident.
alternative	Alternative information about the incident.
alternative_txt	Text description of the alternative information.
multiple	Indicates whether the incident involved multiple simultaneous attacks.
success	Indicates whether the terrorist act was successful.
suicide	Indicates whether the incident involved a suicide attack.
attacktype1	Numeric code for the primary attack type.
attacktype1_txt	Text description of the primary attack type.
targtype1	Numeric code for the general target type.
targtype1_txt	Text description of the general target type.
targsubtype1	Numeric code for the more specific target subtype.
targsubtype1_txt	Text description of the specific target subtype.
corp1	Name of the corporation or entity targeted.
target1	Specific description of the target.
natlty1	Numeric code for the nationality of the target.
natlty1_txt	Text description of the nationality of the target.
gname	Name of the terrorist group responsible for the incident.
motive	Motivation or reason behind the incident.
guncertain1	Indicates uncertainty about the involvement of a gun.
individual	Indicates whether the incident was carried out by an individual.
nperps	Number of perpetrators involved in the incident.
nperpcap	Number of hostages or kidnap victims.
claimed	Indicates whether a group claimed responsibility for the incident.
weaptype1	Numeric code for the general type of weapon used.
weaptype1_txt	Text description of the general type of weapon used.
weapsubtype1	Numeric code for the more specific subtype of weapon used.
weapsubtype1_txt	Text description of the specific subtype of weapon used.
weapdetail	Detailed description of the weapon used.
nkill	Number of total confirmed fatalities.
nkillus	Number of U.S. citizens killed.
nkillter	Number of terrorists killed.
nwound	Number of totals confirmed non-fatal injuries.
nwoundus	Number of U.S. citizens wounded.
nwoundte	Number of terrorists wounded.
property	Indicates property damage in the incident.
propextent	Extent of property damage.
propextent_txt	Text description of the extent of property damage.
propvalue	Value of property damage.
propcomment	Comments on the property damage.

ishostkid	Indicates whether hostages were taken.
ransom	Amount of ransom demanded, if applicable.
addnotes	Additional notes or comments.
dbsource	Sources providing information about the incident.

In the article “Prediction of Future Terrorist Activities Using Deep Neural Networks” [33], they propose the use of 34 attributes to predict the suicide and success. These were iyear, imonth, iday, extended, provstate, latitude, longtitude, specificity, vicinity, crit1, crit2, crit3, doubtterr, multiple, ishostkid, propextent, ishostkid, ransom, country, city, gname, individual, nkillus, nkiller, nwound, nwoundus, nwoundte, ishostkid, targtype1, suicide, success, weaptype1, region and attacktype1. In [35] they proposed 14 features as the most important ones: extended, latitude, longitude, success, suicide, nkill, propextent, nwound, country, region, city, attacktype1, targtype1 and weapontype1 in order to classify the danger level of an attack. In another attack success prediction problem [30], only 5 main features were used: suicide, success, weapon type, region, and attack type. In [57] 23 characteristics were selected for the study on the prediction of terrorist targets, including iyear, imonth, iday, country, region, latitude, longitude, nkill, nwound, gname, and crit1-crit3 (entry criterion), doubtterr (suspected terrorism), guncertain1, property, propextent (extent of property damage), attacktype1, weapsubtype1, targtype1, targsubtype1, claimed (claim of responsibility), claimmode. In another project [58], they used 9 features to predict the success of an attack: country, region, attack type, target type, weapon type, group name, suicide, multiple and ishostkid. In [26], 8 features well selected iyear, city, country, gname, weapontype1, ransom, summary, ishostkid to categorize a terrorist attack. Lastly, we can see that in this paper [59] 30 features were used to : provstate, latitude, longtitude, specificity, vicinity, crit1, crit2, crit3, success, suicide, attacktype1, targtype1, targsubtype1, ishostkid, ishostkid, ishostkid, weapontype1, weaponsubtype1, nkill, nkillus, nkiller, nwound, nwoundus, nwoundte, property, ishostkid, INT_LOG, INT_IDEO, INT_MISC, INT_ANY while trying to predict the type of the attack. For our study, after considering our own dataset and target along with the papers mentioned previously, made some decisions to merge some columns. For example, we created a new feature called “casualties” where we added the number of deaths and the number of wounded caused by an attack. In total, we concluded on the following starting 31 features (see Table 5) to be the dataset we will analyze and pre-process. In Figure 17 the feature correlation is displayed, which is a first look to understand how these features affect one another and what information we can extract from their relations. For example, we see that weapon type and attack type have a strong connection between them, like country and nationality as well.

Next step was the data cleaning. The first thing we did is to see if we got any missing data and figure out how to deal with null values. The columns with missing were city, provstate, latitude, longitude, specificity, doubtterr, targsubtype1, target1, weapsubtype1, guncertain1, nkill, nwound, natlty1, ransom, propextent, ishostkid. Some of these features are numerical, others are textual and others categorical, we treated each case differently. In the case of city, provstate and target1, we filled the missing data with the Unknown label, that already exists in our dataset. We also searched for any negative values inside our numeric features to replace them with 0 as we account them as errors. For the rest of our features, we applied the most frequent strategy

provided by SimpleImputer. Next, we dealt with converting the textual data to numeric values. Processing features with text data in NN or DNN is not feasible directly. However, several techniques can be employed to convert text data into numeric representations, such as TFIDF, Word2Vec, GloVe, One-hot encoding, etc. This paper [30] utilizes the LabelEncoder class from the sklearn library to transform non-numeric data into numeric data. This choice is made because labels are hashable and can be compared to numerical labels.

Here is a sample code on how we dealt with the negative values:

```
columns_with_nulls = null_counts[null_counts > 0].index
df = df[numer_features].applymap(lambda x: max(0, x) if pd.api.types.is_numeric_dtype(x) else x)
```

Here, as we said, we fill our missing data:

```
data.loc[:, 'city'].fillna('Unknown', inplace=True)
data.loc[:, 'provstate'].fillna('Unknown', inplace=True)
data.loc[:, 'target1'].fillna('Unknown', inplace=True)
data.loc[:, 'gname'].fillna('Unknown', inplace=True)

imputer = SimpleImputer(strategy='most_frequent')
df = pd.DataFrame(imputer.fit_transform(data), columns = data.columns)
```

And here we transformed our text data to numerical:

```
lebel_encoder = preprocessing.LabelEncoder()
df['city'] = lebel_encoder.fit_transform(df['city'])
df['provstate'] = lebel_encoder.fit_transform(df['provstate'])
df['target1'] = lebel_encoder.fit_transform(df['target1'])
df['gname'] = lebel_encoder.fit_transform(df[' gname' ])
```

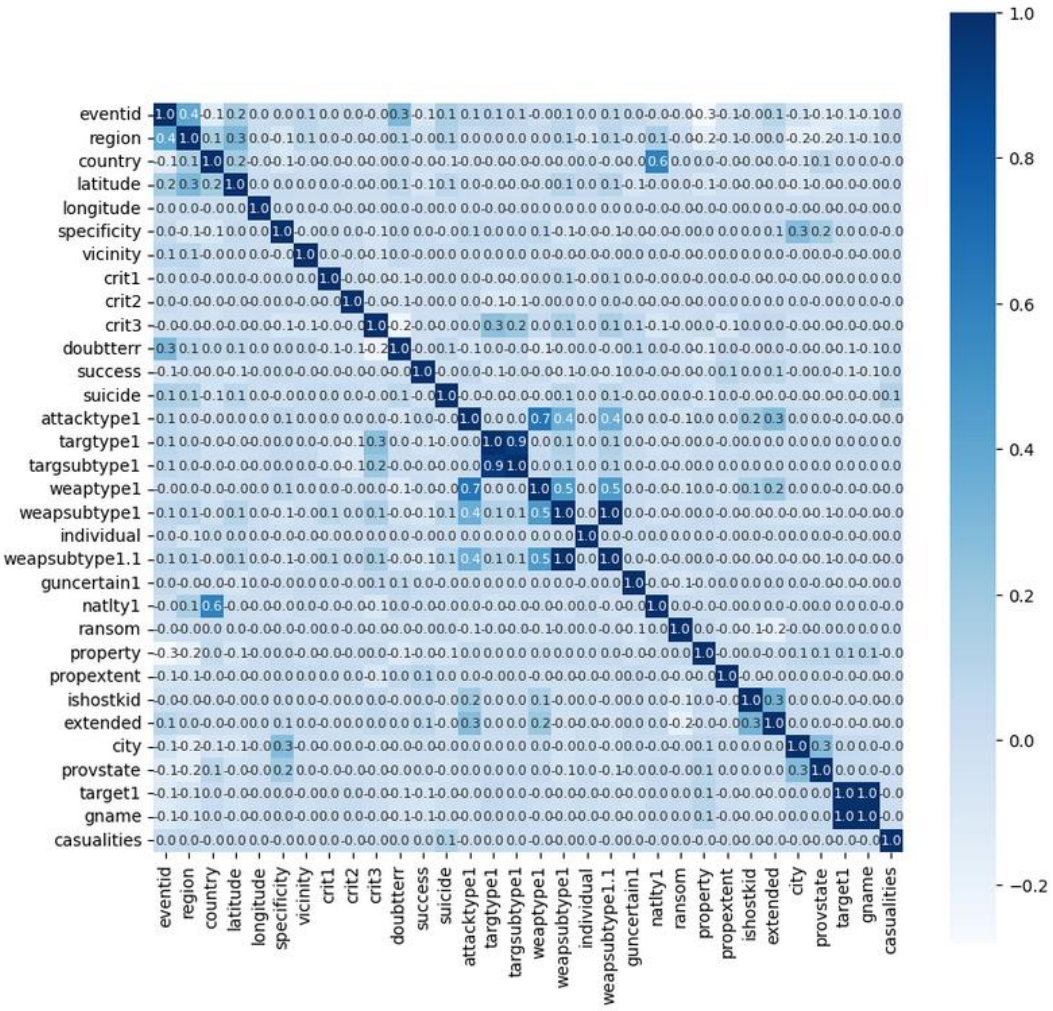


Figure 17: GTD Feature correlation.

Table 5: GTD Feature List

Feature Name		
iyear	targtype1	latitude
region	targsubtype1	Longitude
Country	individual	weaptype1
City	target1	weapsubtype1
Provstate	specificity	guncertain1
casualties	Vicinity	gname
crit1	natly1	Propextent
crit2	Ransom	doubtterr
crit3	property	ishostkid
success	Suicide	extended
attacktype1		

Normalization is a crucial step in handling diverse data ranges within the GTD. Certain columns may contain binary values (0 and 1), while others exhibit values in the hundreds or thousands. This disparity can pose challenges for learning algorithms, hindering their ability to discern patterns and converge to a global minimum. Therefore, it becomes imperative to normalize the data before subjecting it to a learning model, ensuring that it falls within the range of 0 to 1 or −1 to 1. In this study, the MinMaxScaler from the sklearn library is employed. This scaler, for each feature value, subtracts the average of all values and divides it by the standard deviation, effectively transforming the data into the range of −1 to 1. The standardization formula is represented by equation (1), where X_i represents all the samples for a given feature, $\bar{X}$ denotes the average of all samples for the feature, and s represents the standard deviation. Using SMOTE: Synthetic Minority Oversampling Technique presented by Chawla et al. in 2002 [36] that is now available as a tool to be used in Python [37] we were able to balance the data. Train-Test Splitting and Classification Phase. Furthermore, we also applied the PCA algorithm for dimensional reduction, Principal Component Analysis (PCA) extracts M-dimensional feature matrices from N-dimensional matrices. We created this way a new dataset with 7 new features, created from the PCA for each of our target data. This is the basic phase of the proposed system where data is divided, and a classification model is built. The data is divided into three groups training, validation, and test 5:3:2, 6:2.5:1.5 and 7:2:1 respectively. It is a significant step that plays an influential role in preparing the data for classification. This division is so important in training ML algorithms to reduce errors and increase accuracy.

$$Z_i = \frac{X_i - \bar{X}}{s} \quad (1)$$

3.3.2 QFactors Dataset

In this section we will explain our second dataset that was provided from the mentioned thesis [22]. We will see some of the features that we mentioned and explained in the previous section and be introduced to new ones (see Table 6). The fact that there was no missing data here, was very helpful. We performed only the previous steps mentioned, to convert text to numbers and normalize the data.

Table 6: QFactors dataset features

Feature		
iyear	natlty1	education
gname	individual	freedom
iyear	weaptype1	gni_per_cap
Extended	weapsubtype1	gov_conf
country	nkill	immigrants
life_exp	Pop_den	pov_gap

In Figure 18, we can see the correlation between all features. We can understand that using these we gain a good amount of information regarding out target data, more from some features, less than others. Going a step forward, we wanted to see what features really affect the outcome of our classification, and if we can achieve descent results using a smaller number of features. For this reason, we applied several machine learning algorithms like DT, RF, and LDA to create a sub-set with the 8 more important features. The results provided by these algorithms, differ from each other, but we can point that there is a common orientation. We can see from Table 7, the results of these algorithms, that the most important overall features, are life expectancy, poverty gap, the number of immigrants, country, longitude, and latitude. We decided to categorize the features that we are going to use in 3 categories: geographic (longitude, latitude), attack-based (attack type, attack target), sociodemographic (life expectancy, education) and grievances (poverty gap, government violence levels) [60].

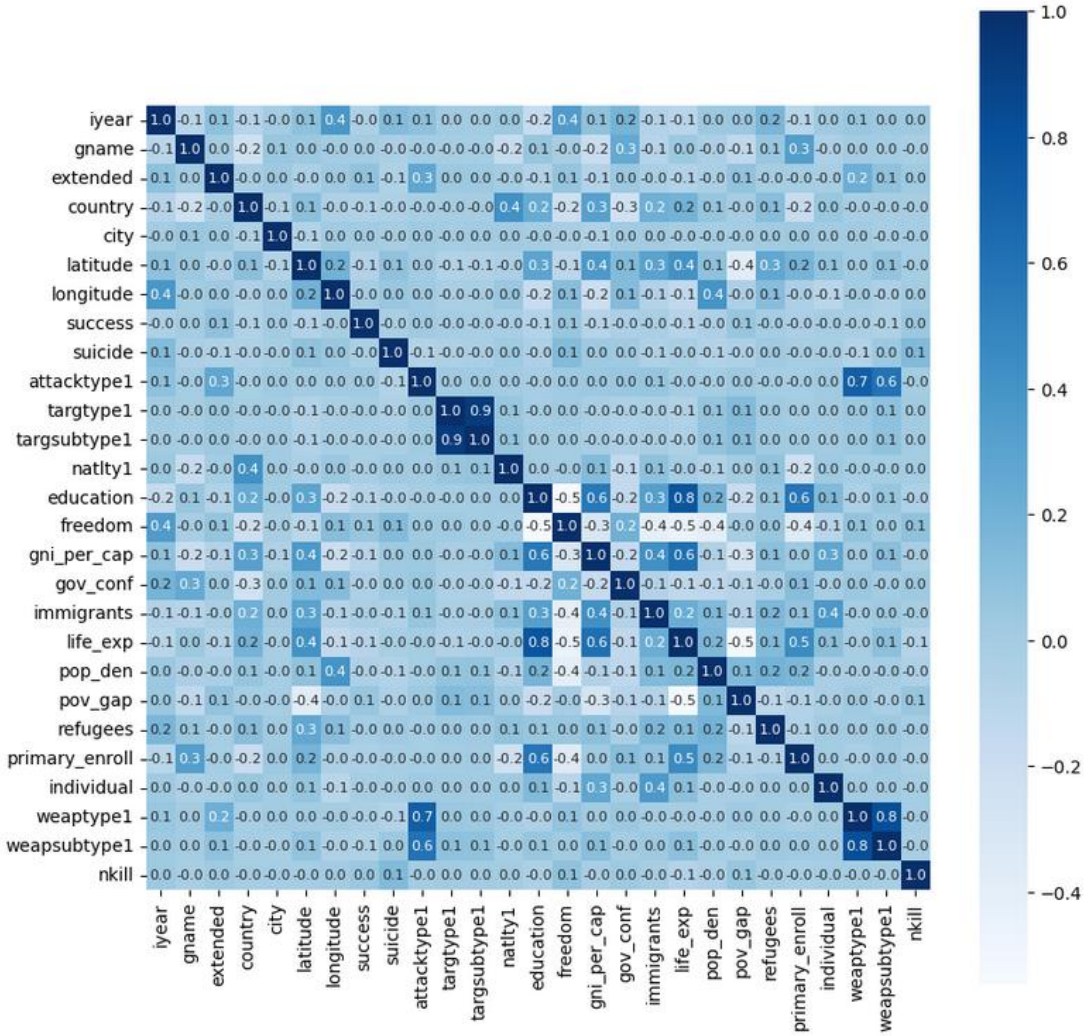
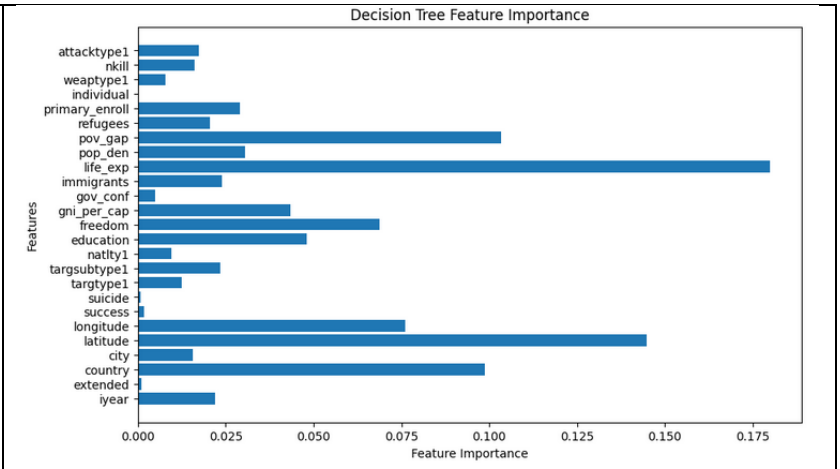
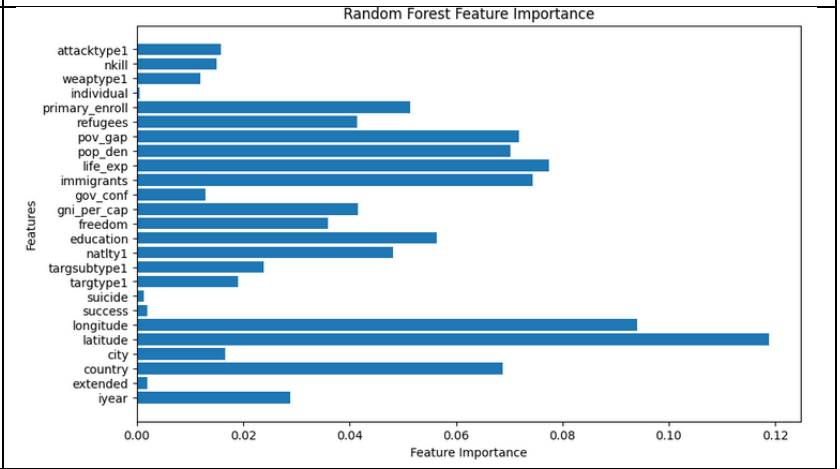
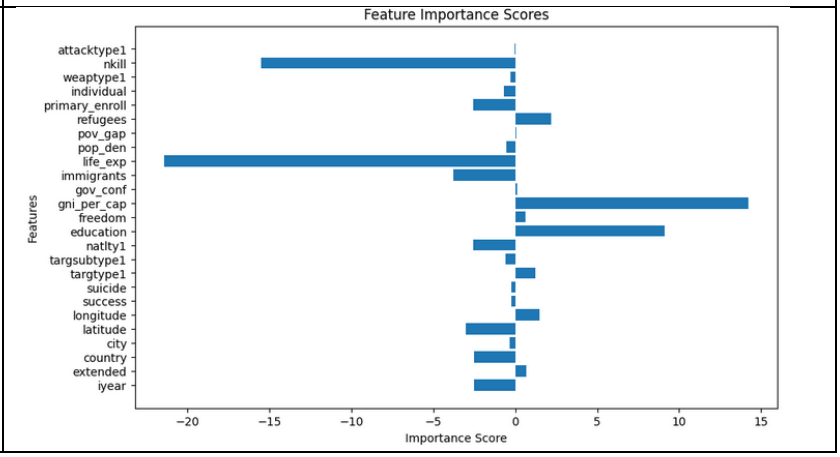


Figure 18: QFactors Features Correlation

Table 7: Feature importance, generated from Decision Tree, Random Forest, and Linear Discriminant Analysis

Decision Tree Classifier	
Random Forest	
Linear Discriminant Analysis	

3.4 Model Structure

We created Multi-Class Classification DNN models and applied different feature sub-sets to compare the results. Following data pre-processing and feature selection, we proceeded to the development of our predictive models for each dataset we created. For our classifications we created various DNN models to compare their performance. In this section, we will outline the architecture of both NN and DNN models. A NN models is an advanced computational model structured as a graph, featuring nodes connected by weighted edges [61] [62] [63]. The basic NN model comprises three layers (see Figure 19), with the initial layer designated as the input layer, responsible for receiving input features. The second layer, termed the hidden layer, and the final layer, known as the output layer, are crucial for making predictions. During the forward propagation process, input data undergoes multiplication with edge weights, and the outcomes are aggregated for each node in the subsequent layer, a step referred to as a weighted sum. Subsequently, this sum undergoes processing through an activation function, introducing non-linearity to facilitate the model's capacity to comprehend and represent intricate data patterns. To train a NN, a weight matrix of the same size as input features is required. As an illustrative example, we will consider 10 features, where 9 serve as input features and 1 as the output feature categorizing predictions into their political label. For training, all data is organized into a table. With a dataset comprising 181,690 instances of terrorist activities, the size of the input matrix represented by X is 181,691 × 9. If there are 10 units in the first layer, the size of the weight matrix becomes 33 × 10. The initialization of these weights is carried out randomly using the Glorot Uniform initializer (1). Additionally, a non-linear function, Rectified Linear Unit (ReLU) (2) [63], contributes to the model's complexity and learning capabilities.

$$Z_1 = W_1^T x\, X + b_1 \quad (1)$$

$$A_1 = ReLU(Z_1) \quad (2)$$

For the output layer, the output of the hidden layer is multiplied by distinct weights. In this scenario, with 10 units in the hidden layer and 6 units in the output layer, the dimensions of the weight matrix become 10 × 6. Additionally, a bias is introduced at this layer to enhance complexity. In the output layer, sparse categorical cross entropy is applied, and is calculated by comparing these predicted values with the actual values. In the backpropagation phase, the loss derivative is computed for both the output and hidden layers, and weights are adjusted using optimization techniques, like Adam [64]. A representative architecture of the NN is depicted in Figure 20, showcasing the input layer housing the input data. These data traverse to the hidden layer, and the output from the hidden layer is then directed to the output layer and the computation is executed and finally, the loss function is computed at the output layer.

A DNN [65] [66] encompasses more layers compared to a single-layer NN. Typically, having more than two hidden layers categorizes a neural network architecture as a DNN. Larger DNNs can feature hundreds of layers, exemplified by ResNet [67] with its impressive 152 layers. Recent advancements in various fields have showcased DNNs achieving state-of-the-art accuracy across diverse applications. Figure 20 presents a schematic representation of a DNN architecture with 4

hidden layers. The process of forward propagation in a DNN mirrors that of an NN. The computation for a single hidden layer in an NN is extended to L − 1 hidden layers in a DNN. During backpropagation, the values of weights of each layer undergo updates using the Adam optimization algorithm. The initialization phase involves setting the weights and biases for all layers using the Glorot Uniform initializer [68]. In the forward propagation step, non-linear activation ReLU [63], is computed at each layer. For the last layer, the sparse categorical cross entropy loss function is applied to calculate the loss. For multiclass classification, softmax [69] is used, while in binary classification scenarios, the sigmoid activation function [70] is employed. During backpropagation, the derivative of the loss function concerning weights and biases is determined for each layer. The weights and biases are then updated using the Adam optimization algorithm [64]. There are also other optimization algorithms like gradient descent with momentum [71], and RMSprop [72]. This paper opts for Adam optimization in the learning process due to its effectiveness in training deep neural networks. Adam optimization involves variables like vcorrected $v_{corrected}^{(t)}\ dW^{[i]}$ and scorrected $s_{corrected}^{(t)}\ dW^{[i]}$, representing the exponentially weighted averages of past gradients with bias correction and the exponentially weighted averages of the squares of past gradients for layer l, respectively.

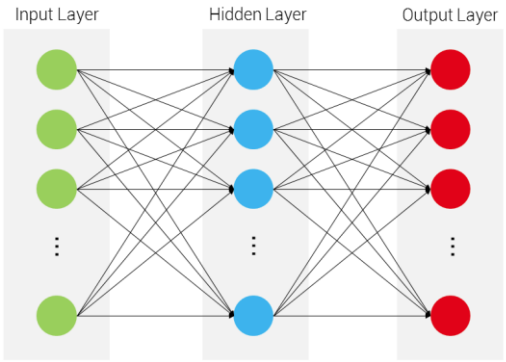


Figure 19: Simple NN model architecture

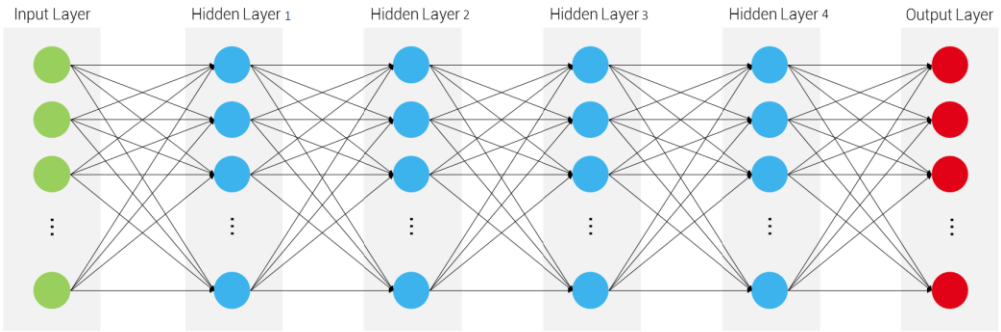


Figure 20: 4 hidden-layers DNN model architecture

3.5 Model Training and Implementation

After the dataset was prepared, training of the DNN models could start. With our model, we want to classify the terrorist groups with their political label. Also, we want to classify the unknown values of our target. For this case, we removed all the terrorist groups with political label marked as unknown and use it for our later prediction. Then we dealt with the unbalanced class by using SMOTE for the rest of the training data. In the end you split our balanced dataset to train, test and validate with the portion of 5:3:2, 6:2.5:1.5 and 7:2:1 respectively.

Here is a sample code, where the data X, is scaled, then SMOTE is applied, and lastly, the data is split into three sub-sets in ratio 7:2:1 to train and test the performance of each DNN model.

```
# scale the dataset for training
X_scaled = scaler.fit_transform(X)

# apply SMOTE to deal with unbalanced classes
smote = SMOTE(sampling_strategy='auto')
X_resampled, y_resampled = smote.fit_resample(X_scaled, y)

# train test split data
X_train, X_test, y_train, y_test = train_test_split(X_resampled, y_resampled, test_size=0.3, random_state=42)

# split test to test and validate data
X_test, X_validate, y_test, y_validate = train_test_split(X_test, y_test, test_size=0.3, random_state=42)
```

Then we created our DNN model with 1 input layer with units equal to the number of features and 4 hidden layers with 100, 50, 30 and 10 units each. For our multiclass DNN model, we used the softmax activation in our output layer, with the output number of units equal to the number of classes we are trying to predict in this case 6. First, we performed this procedure to the dataset containing all our features. Later, through our feature selection, we created a sub-set containing 8 features that we chose as the most important one for our target.

```
model = Sequential()

# Input layer
model.add(Dense(units=X_train.shape[1], activation='relu', input_shape=(X_train.shape[1],)))

# First hidden layer with 100 units and ReLU activation
model.add(Dense(units=100, activation='relu', kernel_initializer=GlorotUniform()))

# First hidden layer with 50 units and ReLU activation
model.add(Dense(units=50, activation='relu', kernel_initializer=GlorotUniform()))

# Second hidden layer with 30 units and ReLU activation
model.add(Dense(units=30, activation='relu', kernel_initializer=GlorotUniform()))

# Third hidden layer with 10 units and ReLU activation
model.add(Dense(units=10, activation='relu', kernel_initializer=GlorotUniform()))

# Output layer with sigmoid activation (assuming multi-class classification)
model.add(Dense(units=y_train.shape[1], activation='softmax'))

# Compile the model
model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])

# Train the model
history = model.fit(X_train, y_train, epochs=50, batch_size=32, validation_data=(X_test, y_test))

# Evaluate the model on the test set
loss, accuracy = model.evaluate(X_test, y_test)
```

Finally, we also applied the PCA method to create a dataset with less features. For our experiments we concluded in the use of 8 number of components for our PCA dataset. In Figure 21, we can see the correlation between these new features created for each of our target.

```
# Create PCA instance
pca = PCA(n_components=8)

# Fit and transform the Test data
X_PCA = pca.fit_transform(X_resampled)

X_PCA = pd.DataFrame(X_PCA)
X_PCA.columns = ['Column1', 'Column2', 'Column3', 'Column4', 'Column5', 'Column6', 'Column7', 'Column8']

X_train, X_test, y_train, y_test = train_test_split(X_PCA, y_resampled, test_size=0.3, random_state=42)

X_test, X_validate, y_test, y_validate = train_test_split(X_test, y_test, test_size=0.3, random_state=42)
```

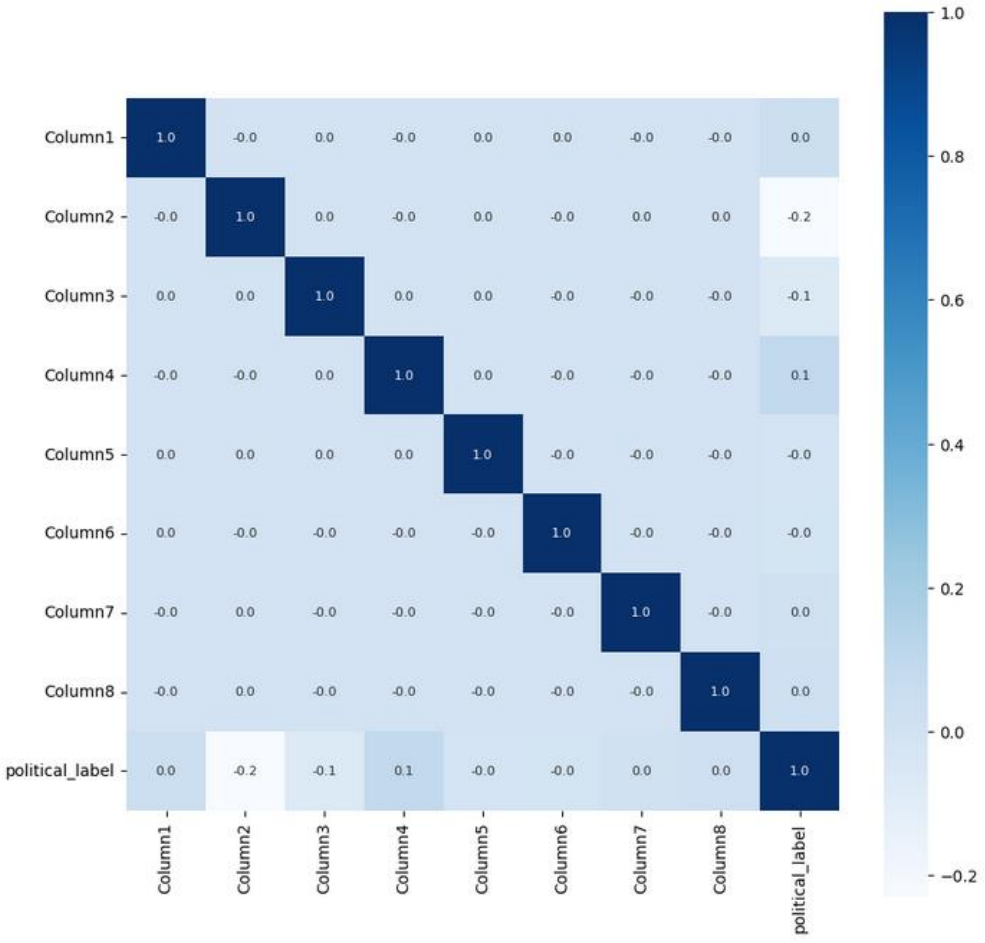


Figure 21: Target Type Correlation PCA

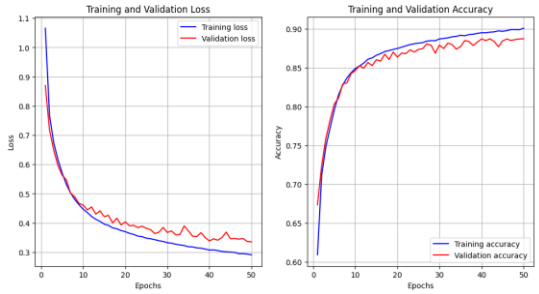
4. Results

In this thesis, to explore the performance of our DNN models, both GTD and QFTD datasets are respectively divided into a training, testing, and validation set at a ratio of 5:3:2, 6:2.5:1.5 and 7:2:1 for different random state numbers. The training set is selected randomly 100 times to construct the classifiers and account for variability in the input data. To compare the model's performance, we measure the result in terms of the test classification precision, which is a criterion for correctly classifying labeled events. All our models run for 50 epochs, each achieving different accuracy scores. Training time for each model also differs. The DNN model showed better results than the Simple NN model and we are going to analyze them in this section.

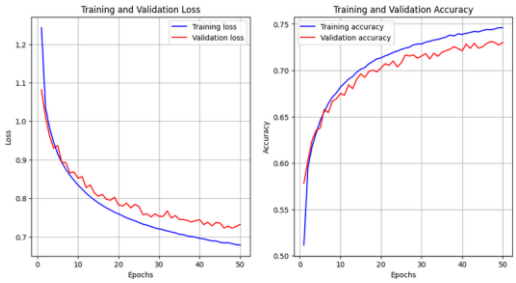
In the learning curve plots Figure 22 and Figure 23, the training and validation loss curves depict the performance of the model during training.

- A decreasing training loss indicates that the model is learning to fit the training data better over epochs.
- The validation loss curve provides insights into the generalization performance of the model. If the validation loss decreases along with the training loss, it indicates that the model is not overfitting.
- Similarly, the training and validation accuracy curves depict the classification accuracy of the model during training.

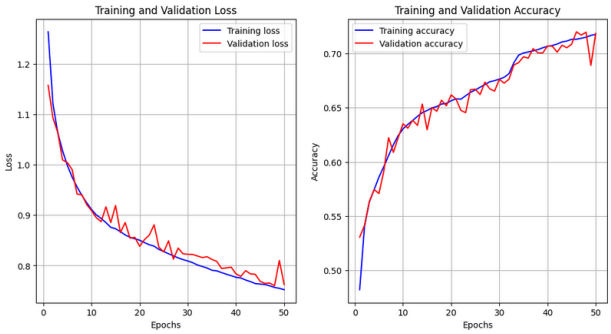
A large gap between the training and validation loss curves may indicate overfitting, while a small gap with both losses decreasing suggests that the model is learning well and generalizing to unseen data effectively. From Figure 22 and Figure 23, we can safely say that there was a smoother learning process in the QFTD, as the validation loss sticks closer to the training loss.



(a) Loss and Accuracy during model training on the full dataset

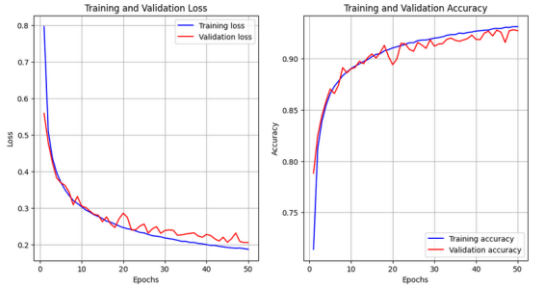


(b) Loss and Accuracy during model training on PCA features

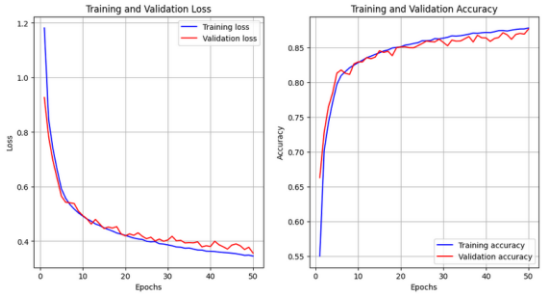


(c) Loss and Accuracy during model training on the selected features

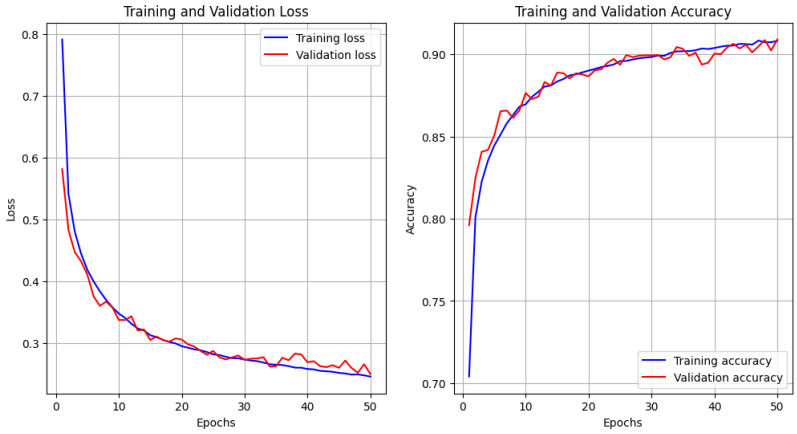
Figure 22: DNN model Loss and Accuracy during training process on GTD



(a) Loss and Accuracy during model training on the full dataset



(b) Loss and Accuracy during model training on PCA features



(c) Loss and Accuracy during model training on the selected features

Figure 23: DNN model Loss and Accuracy during training process on QFTD

Table 8: Model training accuracy summary for each sub-set

GTD		
Full dataset (30 features) Test Loss: 0.3495 Test Accuracy: 0.88 Training took 13.38 minutes.	**PCA** Test Loss: 0.7323 Test Accuracy: 0.7300 Training took 9.71 minutes.	**Selected features** Test Loss: 0.7624 Test Accuracy: 0.7185 Training took 12.82 minutes.
F1-score: Accuracy: 0.89 macro avg: 0.89 weighted avg: 0.89	**F1-score:** Accuracy: 0.73 macro avg: 0.73 weighted avg: 0.73	**F1-score:** Accuracy: 0.72 macro avg: 0.71 weighted avg: 0.71
QFTD		
Full dataset (25 features) Test Loss: 0.2057 Test Accuracy: 0.93 Training took 11.38 minutes.	**PCA** Test Loss: 0.3564 Test Accuracy: 0.88 Training took 9.71 minutes.	**Selected features** Test Loss: 0.2361 Test Accuracy: 0.91 Training took 7.11 minutes.
F1-score: Accuracy: 0.93 macro avg: 0.93 weighted avg: 0.93	**F1-score:** Accuracy: 0.88 macro avg: 0.88 weighted avg: 0.88	**F1-score:** Accuracy: 0.91 macro avg: 0.91 weighted avg: 0.91

The DNN models were trained for 50 epochs and in Table 8 we can see the mean accuracy summary of over 100 repetitions for the different features of each dataset. It can be seen that the DNN models demonstrated better performance on the QFTD, suggesting that there is a link between demographic, social, and economic factors in a country influencing specific terrorist group behavior. The best accuracy on GTD was achieved using all 30 features, reaching 88% accuracy score. Both PCA sub-sets had lesser results, 73% for GTD and 88% for QFTD. On the other hand, using the QFTD, the model achieved more than 90% accuracy in both selected features and the whole dataset, 91% and 93% respectively. The difference between them is only 2%, suggesting that our selected feature subset, indeed has an effect on prediction accuracy. Another important fact is the time difference between the two models, as the model trained on the QFTD with the selected features, had the fastest training time. The formulae to calculate accuracy, precision, recall, and F1-Score are given in the following equations.

$$\text{Accuracy} = \frac{TN+TP}{TN+TP+FN+FP}$$

$$\text{Precision} = \frac{TP}{TP+FP}$$

$$\text{Recall} = \frac{TP}{TP+FN}$$

$$F1 - Score = 2 \text{ x } \frac{precision \text{ x } recall}{precision+recall}$$

TP stands for true positive, TN means true negative, FP means false positive, and FN means false negative. All these experiments indicate that DNN trained with the QFTD, is able to achieve an accuracy of more than 90% in both train and test datasets. The comparison of precision, recall, and F1-Score in test data computed by DNN is given in Table 8. It can be observed that DNN has achieved more than 91% in precision, recall, and F1-Score. This is another demonstration that as the number of layers increases, the network is able to learn the features in the dataset more effectively and make more accurate predictions.

Based on the provided precision, recall, and F1-score values for the six different classes (Islamist, Left-wing, Separatist, Other, Right-wing, Anti-colonial) in data.

, we can draw the following interpretations:

1. **Precision**:
    - Precision measures the accuracy of the positive predictions made by the classifier.
    - A precision value of 1.00 for the Islamist class indicates that all instances predicted as Islamist were Islamist.
    - Among the other classes, Left-wing and Separatist also show high precision values, indicating that the classifier has a low false positive rate for these classes.
2. **Recall**:
    - Recall (also known as sensitivity) measures the ability of the classifier to correctly identify positive instances out of all actual positive instances.
    - The Islamist class has a recall value of 1.00, indicating that all actual Islamist instances were correctly identified by the classifier.
    - Other classes like Right-wing and Anti-colonial also show high recall values, suggesting that the classifier effectively identifies these classes.
3. **F1-score**:
    - The F1-score is the harmonic means of precision and recall and provides a balanced measure of a classifier's performance.
    - The Islamist class has a perfect F1-score of 1.00, indicating both high precision and recall.
    - Other classes like Left-wing and Separatist also have relatively high F1-scores, suggesting a good balance between precision and recall.

Overall, these evaluation metrics indicate that the classifier performs well across multiple classes, particularly for Islamist, Left-wing, and Separatist classes, which have high precision, recall, and F1-score values. However, the performance may vary depending on the specific class, and further analysis may be needed to identify any potential areas for improvement in classification accuracy.

Table 9: DNN accuracy scores on test data

GTD FULL						
	Islamist	Left-wing	Separatist	Other	Right-wing	Anti-colonial
precision	1.00	0.88	0.88	0.85	0.84	0.87
recall	1.00	0.90	0.88	0.80	0.90	0.82
f1-score	1.00	0.89	0.88	0.82	0.87	0.84
GTD PCA						
	Islamist	Left-wing	Separatist	Other	Right-wing	Anti-colonial
precision	0.95	0.68	0.76	0.65	0.73	0.62
recall	0.99	0.70	0.64	0.64	0.78	0.64
f1-score	0.97	0.69	0.70	0.65	0.75	0.63
GTD SELECTED FEATURES						
	Islamist	Left-wing	Separatist	Other	Right-wing	Anti-colonial
precision	0.97	0.68	0.64	0.70	0.65	0.67
recall	0.98	0.75	0.76	0.46	0.72	0.62
f1-score	0.98	0.71	0.69	0.72	0.68	0.65
QFTD FULL						
	Islamist	Left-wing	Separatist	Other	Right-wing	Anti-colonial
precision	1.00	0.94	0.94	0.87	0.89	0.93
recall	1.00	0.88	0.94	0.92	0.95	0.86
f1-score	1.00	0.91	0.94	0.89	0.92	0.89
QFTD PCA						
	Islamist	Left-wing	Separatist	Other	Right-wing	Anti-colonial
precision	1.00	0.90	0.86	0.86	0.82	0.86

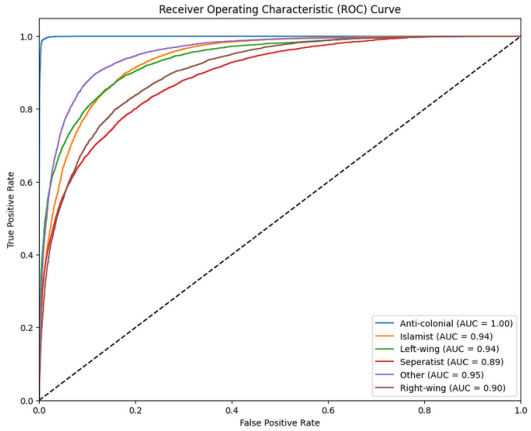
recall	0.99	0.84	0.90	0.81	0.91	0.85
f1-score	0.99	0.87	0.88	0.83	0.86	0.85
QFTD SELECTED FEATURES						
	Islamist	Left-wing	Separatist	Other	Right-wing	Anti-colonial
precision	1.00	0.90	0.93	0.86	0.86	0.91
recall	1.00	0.91	0.92	0.87	0.90	0.86
f1-score	1.00	0.90	0.92	0.86	0.88	0.88

In addition to the classification performance metrics, we can analyze the ROC curve plot diagram to further evaluate the classifier's performance Figure 24 and

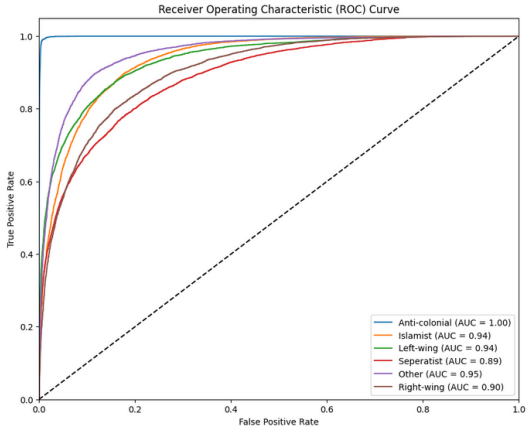
Figure 25. The ROC curve visually represents the trade-off between the true positive rate (sensitivity) and the false positive rate (1-specificity) across different threshold levels.

- The ideal ROC curve would hug the top-left corner of the plot, indicating high sensitivity and low false positive rate across all threshold levels.
- A classifier with a higher Area Under the Curve (AUC) value indicates better overall performance in distinguishing between the classes.
- By analyzing the ROC curve plot, we can assess the classifier's ability to discriminate between different classes and identify the optimal threshold for classification.

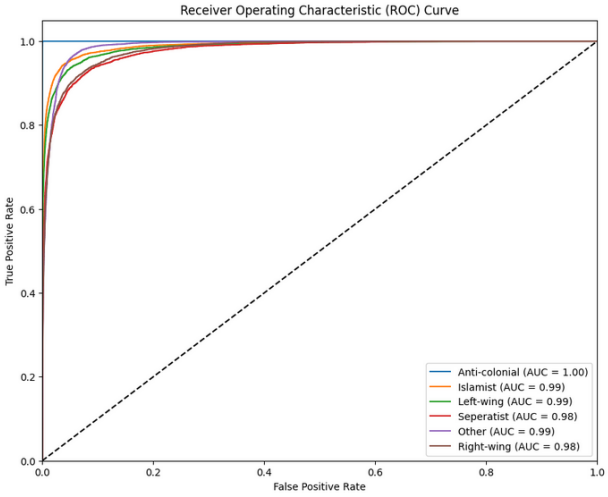
Combining both the classification performance metrics and the analysis of the ROC curve, a comprehensive understanding of the classifier's effectiveness in multi-class classification tasks can be demonstrated.



(a) ROC Curve on PCA dataset

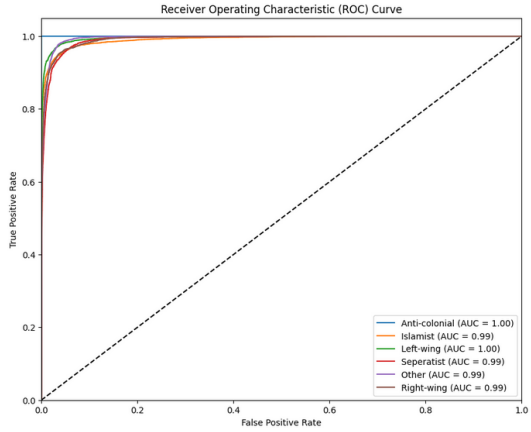


(b) ROC Curve on Selected features (8 features) dataset

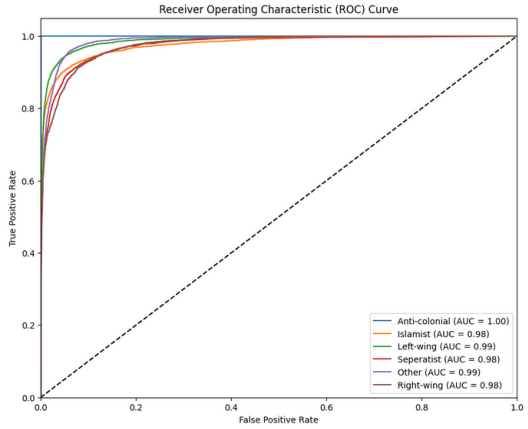


(c) ROC Curve on full dataset (30 features)

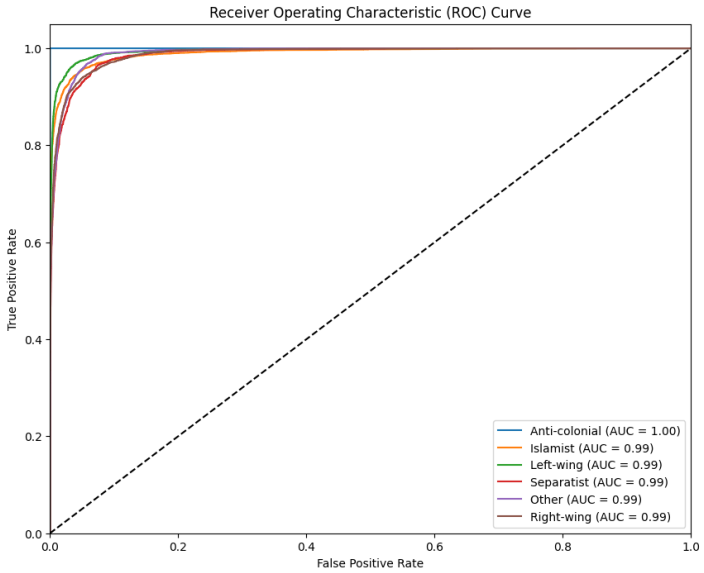
Figure 24: ROC Curve on GTD dataset



(a) ROC Curve on full dataset (25 features)



(b) ROC Curve on PCA dataset



(c) ROC Curve on Selected features (8 features) dataset

Figure 25: ROC Curve on QFTD dataset.

Bellow on Figure 26, the true – false values for the test sub-set are displayed. The DNN models, trained on the QFTD, show better results than the ones from GTD, predicting the Islamist label, close to 100%. Furthermore, the "Other" label, has some wrong classifications as "Left-wing" or "Right-wing", maybe due to the abstractness of the label. Anti-colonials seem to have the least accuracy rate, probably due to low training instances.

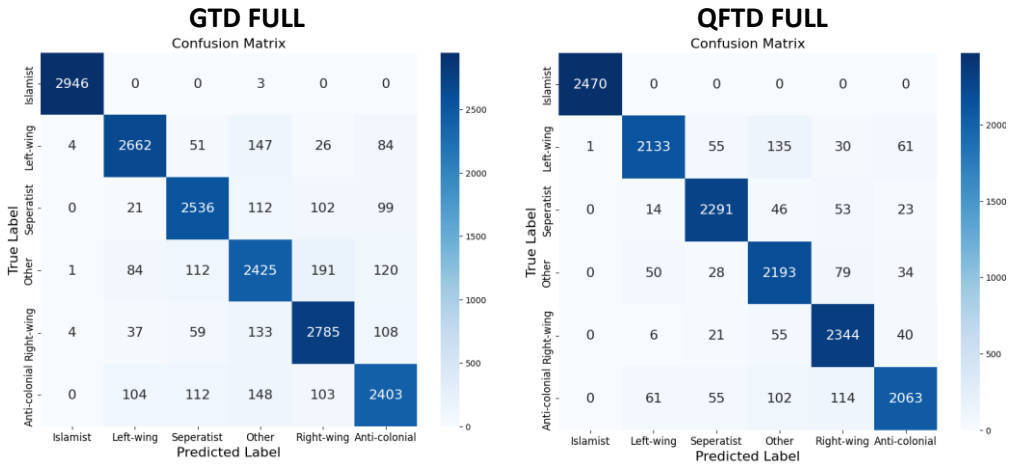


Figure 26: DNN predictions confusion matrix on test data.

4.1 Implementation

Using the previously trained models, it is attempted to classify the "unknown" political label groups. In Figure 27 and Figure 28 we can explore the classification made on the unknown values using each model. We can that the predictions of the models trained on the QFTD, follow the same distribution as our datasets. The most predicted labels are Islamist, followed by left-wing and separatists. Considering the accuracy that we analyzed previously, these results suggest that the QFTD DNN models, are legitimate. This was made to classify the unknown groups and incorporate further information into the databases. Though these may not necessarily correspond to the correct label, given the accuracy of the proposed models, it can be assumed with relatively high confidence that these classes match the actual labels. To validate this classification, further research needs to be done, to accurately define the true label. Regarding the predictions on the GTD values, in all three prediction sets, Separatists, are the 2nd most predicted group, and Anti-

colonial is the least. The most Islamist predictions were made form the model trained in all 30 features. Fourth and fifth places are challenged from different groups each time, but the difference between the number of predicted values is very little.

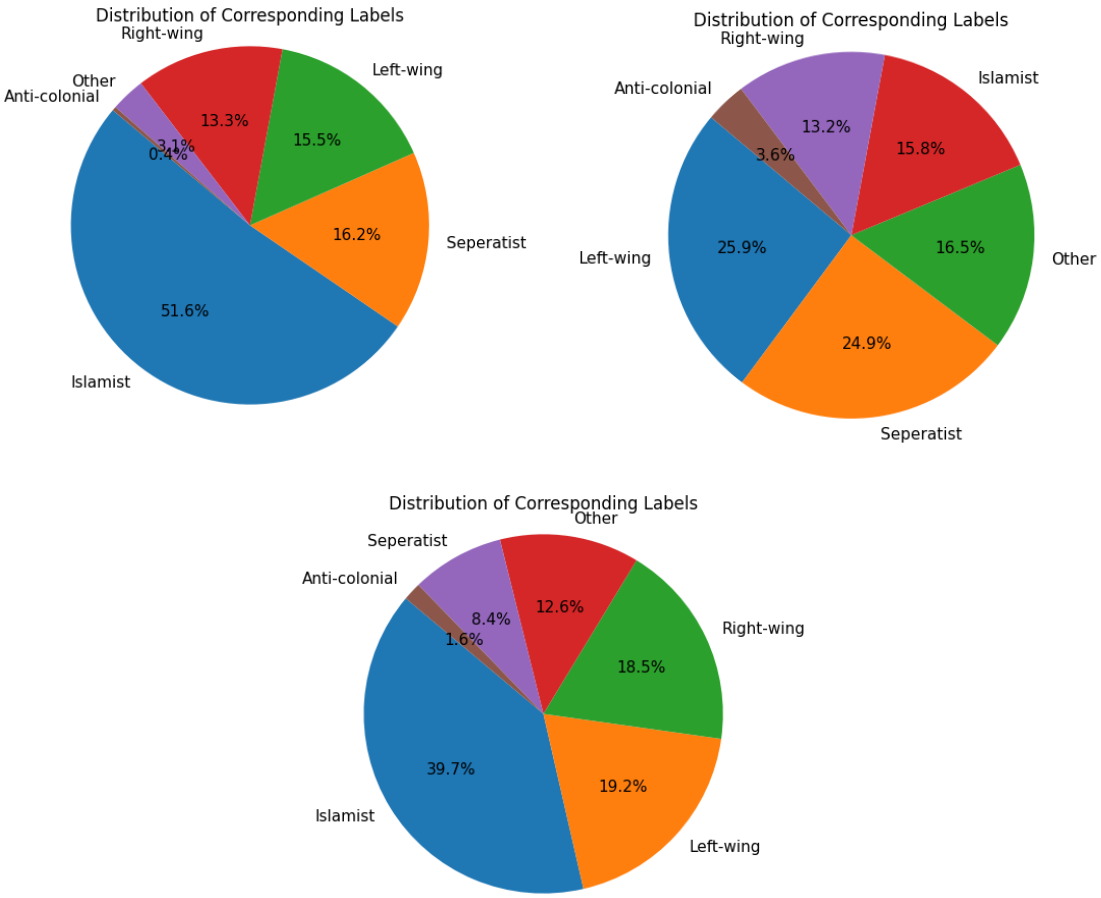


Figure 27: DNN predictions on Unknown labels of GTD dataset.

In the QFTD, Left-wing group, has the 2nd most predictions. All three models had the Anti-colonial group as the least predicted and the Right-wing group is always in third place. Predictions using only the selected features are very close to the ones made use all the features to train the model. This indicates that some of these predictions might be correct.

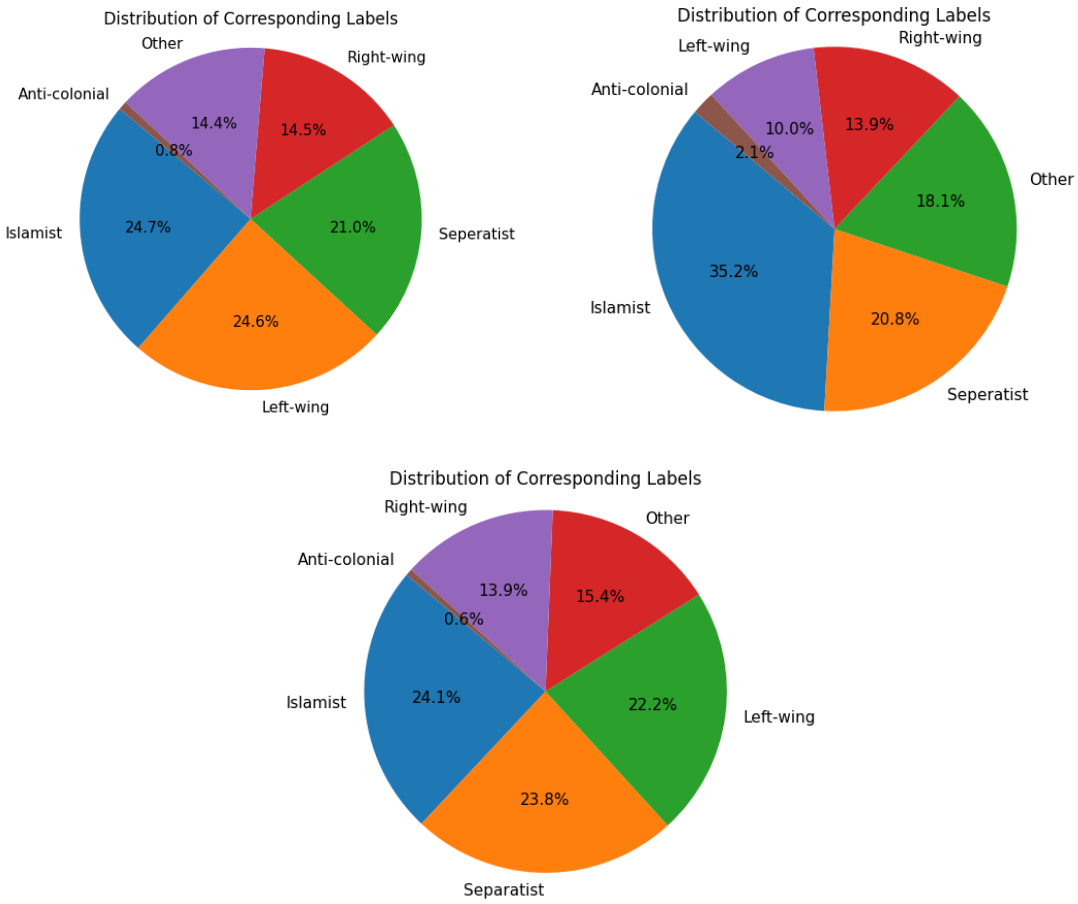


Figure 28: DNN predictions on Unknown labels of QFTD dataset.

4.2 Discussion

From the experiments conducted, it becomes evident that numerous factors play pivotal roles in shaping the performance of a Deep Neural Network (DNN) model. Among these factors are the dataset's length, the dimensionality represented by the number of features, the architecture's complexity denoted by the number of hidden layers, and the extent of training represented by the number of epochs. Moreover, a significant factor influencing the model's efficacy is the distribution of entries across different target classes. In our case, Islamist group has the most recorded attacks, and so it can be seen to be the most accurately predicted class (Table 9). This is why applying techniques like SMOTE, prove indispensable for addressing class imbalances and curbing overfitting tendencies. It is also essential to understand that terrorism is a phenomenon composed of factors of different natures. The results of this thesis also validate this statement, as it was shown, that the models performed better when trained by the QFTD, which included geopolitical information. An important factor is the training time, which was faster on the selected features, pointing out the need to find the importance of the features towards the target data,

and select those that are more suitable. In our case, features encompassing geographic attributes (e.g., longitude, latitude), attack-related characteristics (e.g., attack type, target), sociodemographic indicators (e.g., life expectancy, education), and grievances (e.g., poverty gap, government violence levels) emerge as crucial contributors to model performance and are thus deemed more appropriate for inclusion in the modeling process.

5. Conclusions

In this thesis, we demonstrated a strategy that utilizes ML models to classify a terrorist attack by the political label of the responsible terrorist group considering a broad range of variables that can be related not only to the outcome of a terrorist attack, but to its general nature. Several pre-processing algorithms were investigated for data cleaning while classification tasks using DNN were studied. The inequality of classes in the target value are handled with techniques like SMOTE. Training a DNN model is not a trivial process as there are no standard ways on how to deal with missing data or how many layers should be used. A lot of experimenting is needed to conclude the most suitable model in each case. Also, the hardware used for the training process is very crucial, as with higher specs, we can analyze more data and create more complicated models to achieve even greater results. Various factors can be explored to try and understand the nature and the cause behind terrorist attacks, as they become more organized.

More specifically, to evaluate and extract the most relevant features different Machine learning models were employed like DT, RF, and LDA classifier. The key features identified include geographic (longitude, latitude), attack-based (attack type, attack target), sociodemographic (life expectancy, education) and grievances (poverty gap, government violence levels). Also, it was observed that in many cases, specific domain knowledge for a target can provide more accurate classification compared to using dimensionality reduction techniques, like PCA. It is also deemed essential to use the most important features, in order to maximize the accuracy to computational time ratio.

The importance of data cleaning and data preprocessing has also been emphasized. Dealing with large data sets with a lot of features can significantly increase the greatly the training process, without necessarily improving accuracy. Very important is the evaluation of the models. Using valid metrics, like f1-score, recall and precision, along with validating the models using unknow data to them, we were able to have a more accurate picture of the performance.

Lastly, the models were developed and used to classify the “Uknown” labels, trying to fill this way, the missing data. These values were not validated, and are just a guess, but keeping in mind the accuracy of our model, some of them might be true. This analysis can be further expanded by applying our new “political_label” feature, to explore the classification of other targets, like the attack type or where the attack is going to take place.

Overall, the attribute “political_label”, suggests a strong connection with geographic data, like other similar researches [22] [37] [33]. By analyzing the political tendencies of each country, and monitoring real time news from different sources, we can gain information about possible terrorist attacks. Also, this attribute can aid in the prediction of the perpetrator groups, which can help to create a more complete understanding of a particular situation. Strong political differences create an unstable environment, susceptible to terrorist attacks. It is essential to explore more factors that might affect terrorism, to counter it.

6. References

[1] N. Yuri, V. Gacía-Díaz, C. Montenegro, C. González, G. Camilo and R. Crespo, "Usage of Machine Learning for Strategic Decision Making at Higher Educational Institutions," *IEEE Access,* vol. 7, pp. 75007-75017, 2019.

[2] A. Krzywicki, W. Wobcke, M. Bain, M. J. Calvo and P. Compton, "Data mining for building knowledge bases: techniques, architectures and applications," *The Knowledge Engineering Review,* vol. 31, no. 2, p. 97–123, 2016.

[3] S. A. Chun, S. Shulman, R. Sandoval and E. Hovy, "Government 2.0: Making connections between citizens, data and government," *Information Polity,* vol. 15, pp. 1-9, 210.

[4] I. Tisland, M. Sodefjed, P. Vassilakopoulou and I. Pappas, The Role of Quality, Trust, and Empowerment in Explaining Satisfaction and Use of Chatbots in e-government, Cham: Springer International Publishing, 2022.

[5] S. Ronan, A. A. Montaser and F. Teddy, "Identity Documents Classification as an Image Classification Problem," in *Image Analysis and Processing - ICIAP 2017*, Cham, Springer International Publishing, 2017, pp. 602--613.

[6] V. Mahor, R. Rawat, S. Telang, B. Garg, D. Mukhopadhyay and P. Palimkar, "Machine Learning based Detection of Cyber Crime Hub Analysis using Twitter Data," *IEEE,* pp. 1-5, 2021.

[7] L. A. Iriberri, "Natural Language Processing and e-Government: Extracting Reusable Crime Report Information," in *International Conference on Information Reuse and Integration*, Las Vegas, NV, USA, 2007.

[8] S. Salloum, R. Khan and S. K., "A Survey of Semantic Analysis Approaches," *Springer,* vol. 1153, 2020.

[9] R. Kowalski, M. Esteve and M. S. Jankin, "Improving public services by mining citizen feedback: An application of natural language processing," *Public Admin,* vol. 98, p. 1011–1026, 2020.

[10] Y. Charalabidis, C. Alexopoulos and E. Loukis, "A taxonomy of open government data research areas and topics," *Journal of Organizational Computing and Electronic Commerce,* vol. 26, pp. 1-2, 2016.

[11] L. A. Zhang, "Research on the Impact of Big Data Capabilities on Government's Smart Service Performance: Empirical Evidence From China," *IEEE,* vol. 9, pp. 50523-50537, 2021.

[12] Z. Corri, J. S. Laura, G. Martha and H. Margaret, "Terrorist critical infrastructures, organizational capacity and security risk," *Safety Science,* vol. 110, pp. 121-130, 2018.

[13] J. a. J.A.Giesecke, "Informing Ex Ante Event Studies with Macro - Econometric Evidence on the Structural and Policy Impacts of Terrorism," *Risk Analysis,* vol. 38, no. 4, pp. 804-825, 2018.

[14] G. Campedelli, M. Bartulovic and K. Carley, "Learning future terrorist targets through temporal meta-graphs," *Sci Rep,* pp. 1-15, 2021.

[15] N. M. Abdalsalam, C. Li, A. Dahou and S. Noor, "A Study of the Effects of Textual Features on Prediction of Terrorism Attacks in GTD Dataset," *Eng. Lett.,* 2021.

[16] N. B. Frey, S. Luechinger and A. Stutzer, "Calculating tragedy: assessing the costs of terrorism," *J. Econ. Surv,* p. 1–24, 2007.

[17] A. Tawadros, "Mapping terrorist groups using network analysis: egypt case study," *J. Humanit. Appl. Soc. Sci.*

[18] S. A. Shah, M. Uddin, N. Zada, F. Aziz, Y. Saeed and A. Zeb, "Prediction of future terrorist activities using deep neural networks," *Complexity,* 2020.

[19] Bansal and J. K. Saini, "Computational techniques to counter terrorism: A systematic survey," *Multimedia Tools Appl,* p. 1–26, 2023.

[20] S. Singh, J. Allanach, H. Tu, K. Pattipati and P. Willett, "Stochastic modeling of a terrorist event via the ASAM system," *IEEE International Conference on Systems, Man and Cybernetics,* vol. 6, pp. 5673-5678, 2004.

[21] R. Alhamdani, M. Abdullah and I. Sattar, "Recommender System for Global Terrorist Database Based on Deep Learning," *International Journal of Machine Learning and Computing,* vol. 8, no. 6, 2018.

[22] A. Peng, An Integrated Machine Learning Approach To Studying Terrorism, Yale, 2018.

[23] T. Bjørgo, For preventing terrorism, Basingstoke: Palgrave Macmilan, 2016.

[24] L. Bosi and S. Malthaner, "For discussing terrorism and political violence," in *Political violence. The Oxford handbook of social movements*, 2015, pp. 440-451.

[25] Donatella, "For left-wing political violence: Della Porta," *Clandestine political violence. Cambridge University Press,* 2013.

[26] H. Mo, X. Meng, J. Li and S. Zhao, "Terrorist event prediction based on revealing data," in *IEEE 2nd International Conference on Big Data Analysis (ICBDA)*, Beijing, China, 2017.

[27] D. Fangyu, G. Quansheng, J. Dong, F. Jingying and H. Mengmeng, "Understanding the dynamics of terrorism events with multiple-discipline datasets and machine learning approach," *PLoS ONE,* no. Understanding the dynamics of terrorism, 2017.

[28] S. Kalaiarasi, A. Mehta and D. Bordia, "Using Global Terrorism Database (GTD) and Machine Learning Algorithms to Predict Terrorism and Threat," vol. 9, no. 1, p. 5995–6000, 2019.

[29] M. Xi, N. Lingyu and S. Jiapeng, "Big data-based prediction of terrorist attacks," *Computers & Electrical Engineering,* vol. 77, pp. 120-127, 2019.

[30] S. Firas and T. Zouheir, "A hybrid deep learning-based framework for future terrorist activities modeling and prediction," *Egyptian Informatics Journal,* no. 23, p. 437–446, 2022.

[31] C. Verma, S. Malhotra and V. Verma, "Predictive modeling of terrorist attacks using machine learning," *International Journal of Pure and Applied Mathematics,* vol. 119, p. 6, 2018.

[32] Z. Li, S. Duoyong, L. Bo, L. Zhanfeng and L. Aobo, "Terrorist group behavior prediction by wavelet transform-based pattern recognition," *Discrete Dynamics in Nature and Society,* vol. 2018, p. 1–16, 2018.

[33] U. M. Irfan, Z. Nazir, A. Furqan, S. Yousaf, Z. Asim, A. S. Syed Atif, A.-K. Mahmoud Ahmad and M. Marwan, "Prediction of Future Terrorist Activities Using Deep Neural Networks," *Hindawi Complexity,* p. 16, 2020.

[34] T. Jani, Predicting success of global terrorist activities, 2019.

[35] J. Yu, Z. Hu, T. Xian and Y. Liu, "Hazard Grading Model of Terrorist Attack Based on Machine Learning," *International Journal of Advanced Network, Monitoring and Controls,* vol. 4, no. 2, 2019.

[36] S. N. Semeh BEN SALEM, "Pattern recognition approach in multidimensional databases: Application to the global terrorism database," *International Journal of Advanced Computer Science and Applications,* vol. 7, no. 8, 2016.

[37] M. H. K. Motaz, H. M. A.-E.-E. Tarek and G. M. A. Soliman, "HYBRID CLASSIFICATION ALGORITHMS FOR TERRORISM PREDICTION in Middle East and North Africa," *International Journal of E merging Trends & Technology in Computer Science,* vol. 4, no. 3, 2015.

[38] A. C. Steven Nieves, "Finding Patterns of Terrorist Groups in Iraq: A Knowledge Discovery Analysis," in *Ninth Latin American and Caribbean Conference, Engineering for a Smart Planet, Innovation, Information*, Medellín, Colombia, 2011.

[39] A. Muhammad and R. Muhammad, "Extracting patterns from Global Terrorist Dataset (GTD) Using Co-Clustering approach," *Journal of Independent Studies and Research,* vol. 13, no. 1, 2015.

[40] K. Vivek, M. Manuel, M. Angelo and L. JooYoung, "A Conjoint Application of Data Mining Techniques for Analysis of Global Terrorist Attacks," in *Proceedings of 6th International Conference in Software Engineering for Defence Applications*, Paolo Ciancarini, Manuel Mazzara, Angelo Messina, Alberto Sillitti, Giancarlo Succi, 2018, p. 146.

[41] O. A. Olabanjo, B. S. Aribisala, M. Manuel and S. W. Ashiribo, "An ensemble machine learning model for the prediction of danger zones: Towards a global counter-terrorism," *Soft Computing Letters,* 2021.

[42] L. Gary and D. Laura, "Introducing the Global Terrorism," *Terrorism and Political Violence,,* vol. 19, no. 2, pp. 181-204, 2007.

[43] Dugan L, " The Making of the Global Terrorism Database and Its Applicability to Studying the Life Cycles of Terrorist," *The SAGE handbook of criminological research methods,* p. 175, 2011.

[44] X. P. a. T. Zhang, "Machine learning-based target prediction for terrorist attacks," *Journal of Physics: Conference Series,* vol. 2577, 2023.

[45] G. E. Robinson, "The four waves of global jihad, 1979–2017.," *Middle East Policy,* vol. 24, no. 3, pp. 70-88, 2017.

[46] R. D. Palmer, "Political terrorism in West Africa," in *In Africa and the War on Terrorism*, Rourledge, 2016, pp. 103-112.

[47] A. J. Ravndal, "Right-wing terrorism and militancy in the Nordic countries: A comparative case study," *Terrorism and Political Violence,* vol. 30, no. 5, pp. 772-792, 2018.

[48] D. D. Porta, "Discursive turns and critical junctures: Debating Citizenship after the Charlie Hebdo Attacks," *Oxford University Press,* 2020.

[49] T. A. Halitovich and P. V. Menshikov, "Terrorism Problem In Latin America: Social Class Specificity Of Region," *European Proceedings of Social and Behavioural Sciences,* 2019.

[50] H. Alvi, "Terrorism in Africa," *Insight Turkey,* vol. 2, no. 1, pp. 111-132, 2019.

[51] R. G. E, "The four waves of global jihad 1979–2017," *Middle East Policy,* vol. 24, pp. 70-88, 2017.

[52] G. Stein, B. Chen, A. S. Wu and K. A. Hua, Decision tree classifier for network intrusion detection with GA-based feature selection, New York, NY, USA: Association for Computing Machinery, 2005.

[53] Landgrebe and S. R. Safavian, "A survey of decision tree classifier methodology," *IEEE Transactions on Systems, Man, and Cybernetics,* vol. 21, no. 3, pp. 660-674, 1991.

[54] H. M. Bjoern, K. B Michael, M. Ralf, H. Uwe, B. Peter, P. Wolfgang and A. H. Fred, "A comparison of random forest and its Gini importance with standard chemometric methods for the feature selection and classification of spectral data," *BMC Bioinformatics,* vol. 10, no. 3, 2009.

[55] W. Huazhen, Y. Fan and L. Zhiyuan, "An experimental study of the intrinsic stability of random forest variable importance measures," *BMC Bioinformatics,* vol. 17, no. 60, 2016.

[56] " Linear Discriminant Analysis. In: Robust Data Mining," *Springer,* 2013.

[57] X. P. a. T. Zhang, "Machine learning-based target prediction for terrorist attacks," *Journal of Physics: Conference Series,* 2023.

[58] T. Jani, "Predicting success of global terrorist activities," 2019.

[59] G. Asma El Kissi and B. A. Najoua Essoukri, "Terrorist Act Prediction Based on Machine Learning: Case Study of Tunisia," in *17th International Multi-Conference on Systems, Signals & Devices*, Tunisie, 2020.

[60] J. A. Ravndal, "Explaining right-wing terrorism and violence in Western Europe: Grievances, opportunities and polarisation.," *European Journal of Political Research,* vol. 57, no. 4, pp. 845-866, 2018.

[61] G. K, "An Introduction to Neural Networks," *CRC Press,* 1997.

[62] Y. LeCun, "Backpropagation Applied to Handwritten Zip Code Recognition," *Neural Computation,* vol. 1, no. 4, pp. 541-551, 1989.

[63] G. E. H. Vinod Nair, "Rectified Linear Units Improve Restricted Boltzmann Machines," in *In Proceedings of the 27th International Conference on International Conference on Machine Learning*, Haifa, 2010.

[64] P. K. Diederik and J. Ba, "Adam: A Method for Stochastic Optimization," 2017.

[65] G. Ian, B. Yoshua and C. Aaron, Deep Learning, Cambridge: MIT Press, 2016.

[66] L. D. a. D. Yu, "Deep Learning: Methods and Applications," *Foundations and Trends® in Signal Processing,* vol. 7, no. 3-4, pp. 197-387, 2014.

[67] M. K. Panda, B. N. Subudhi, T. Veerakumar and V. Jakhetiya, "Modified ResNet-152 Network With Hybrid Pyramidal Pooling for Local Change Detection," *IEEE Transactions on Artificial Intelligence,* pp. 1-14, 2023.

[68] A. Diego and O. Fuentes, "Improving Weight Initialization of ReLU and Output Layers," in *Lecture Notes in Computer Science*, 2019, pp. 170-184.

[69] R. Yazhou, Z. Peng, S. Yongpan, Y. Dezhong and X. Zenglin, "Robust softmax regression for multi-class classification with self-paced learning," *Proceedings of the 26th International Joint Conference on Artificial Intelligence,* p. 2641–2647, 2017.

[70] H. Pratiwi al. et, "Sigmoid Activation Function in Selecting the Best Model of Artificial Neural Networks," *Journal of Physics: Conference Series,* 2020.

[71] N. Qian, "On the momentum term in gradient descent learning algorithms," *Neural Networks,* vol. 12, no. 1, pp. 145-151, 1999.

[72] Z. Fangyu, S. Li, J. Zequn, Z. Weizhong and L. Wei, "A Sufficient Condition for Convergences of Adam and RMSProp," *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition,* pp. 11127-11135, 2019.

[73] S. Ruder, "An overview of gradient descent optimization algorithms," 2016.

[74] Al-Mushayt and S. Omar, Automating E-Goverment Services With Artificial Intelligence, Abha 61421, Saudi Arabia: Department of MIS, King Khalid University, 2019.

[75] X. Dongpo, Z. Shengdong, Z. Huisheng and P. M. Danilo, "Convergence of the RMSProp deep learning method with penalty for nonconvex optimization," *Neural Networks,* vol. 139, pp. 17-23, 2021.

[76] Eick and S. A. Aigbe, "Learning Domain-Specific Word Embeddings from COVID-19 Tweets," in *IEEE International Conference on Big Data (Big Data)*, Orlando, FL, USA, 2021.