Terror Attack Classification with the Application of Orange Data Mining Tool and Neo4j Sandbox

Ankit Raj, Suchitra A. Khoje, and Sagar Bhilaji Shinde

Abstract There is no universally accepted definition of terrorism. Terrorism and its ramifications have every once in a while caused massive death and destruction around the world. Current cutting-edge technologies, such as machine learning and deep learning, can predict and classify such attacks efficiently. The major difficulties observed in implementing these strategies are a lack of consistent and clean data, as well as programming knowledge in Python and R. Inconsistent data can be resolved by incorporating graph database features into the dataset, and Python programming can be replaced with the orange data mining tool. As a part of data processing and manipulation software, orange data mining tool employs a machine learning model in a non-coding context. This research study has attempted to replicate the results by using the orange tool and Neo4j Sandbox. In this study, a non-coding approach was used to classify terror attacks by using the orange data mining tool, and the use of graph embeddings as dataset features have assisted in eliminating the problems associated with inconsistent data. The dataset was then subjected to machine learning techniques such as Random Forest, Decision Tree, Support Vector Machine, Naive Bayes, Gradient Boosting, KNN, and Adaboost to classify the terror attacks. Random Forest and Gradient Boosting are the models that can achieve an accuracy score, recall, precision, and F1 score greater than 90%.

Keywords Neo4j Sandbox · Orange data mining tool · GTD · Machine learning · Graphs

A. Raj (✉) · S. A. Khoje
MIT–World Peace University, Pune, Maharashtra, India
e-mail: reachankitat@gmail.com

S. A. Khoje
e-mail: suchitra.khoje@mitwpu.edu.in

S. B. Shinde
NMVPM's Nutan College of Engineering and Research, Pune, Maharashtra, India

J. Hemanth et al. (eds.), *Intelligent Cyber Physical Systems and Internet of Things*,
Engineering Cyber-Physical Systems and Critical Infrastructures 3,
https://doi.org/10.1007/978-3-031-18497-0_6

1 Introduction

Relational and non-relational databases are the two main types of databases. A non-relational form of database is known as graph database. A graph database or graph is a higher dimensional data representation where nodes and relationships are used instead of rows and columns [1]. While the characteristics of those nodes are the rows of the relational database, which indicate the number of entries in a dataset and the nodes in a graph are the entities that represent a column or attribute of the relational database [2]. The global terrorism database managed by the University of Maryland is the dataset used in the proposed research study. Working in the field of machine learning requires a strong working knowledge of the python programming language. Without any python programming experience, Orange Tool provides the flexibility to work in the domain of machine learning [3]. Here, GTD is used to develop a graph database for the proposed project. This graph database has millions of relationships between its thousands of nodes. The graph data science library found in the Neo4j Sandbox plugin was used to calculate certain properties of the graph database, including degree, centrality, and node embedding. Seven machine learning models, including decision tree, random forest, gradient boosting, KNN, SVM, Naive Bayes, and Adaboost were applied to the dataset. According to different performance metrics like AUC Score, accuracy, recall, and F1 score, the best model will be selected. The prediction results were displayed by using the confusion matrix in the orange tool [3].

1.1 Dataset

The University of Maryland-owned global terrorism database remain as the source for research data. The dataset is a compilation of every act of terrorism that has taken place around the globe between the year 1970 and 2019 in a relational dataset format. The attributes in the dataset include the timing and location of the assault, the type of weapon used, target type, causation, and more. There are 136 attributes in the dataset with two lakh entries of the terror incidents [4]. We sorted and filtered the data set due to computing resource constraints and selected 10 instances per year from 1970 to 2020. Consequently, there were 500 records in the sample dataset. Figure 1 shows the geographical spots on the world map, where terror attacks have occurred in the past. It can be seen that the South Asia region is the most terror attack-prone region on the globe [5]. The event İD, event location, event time, event date, event day, event month, event year, longitude, latitude, specificity, proximity, attack type, target type, gun type, weapon type, and others are the attributes of the GTD dataset [5].

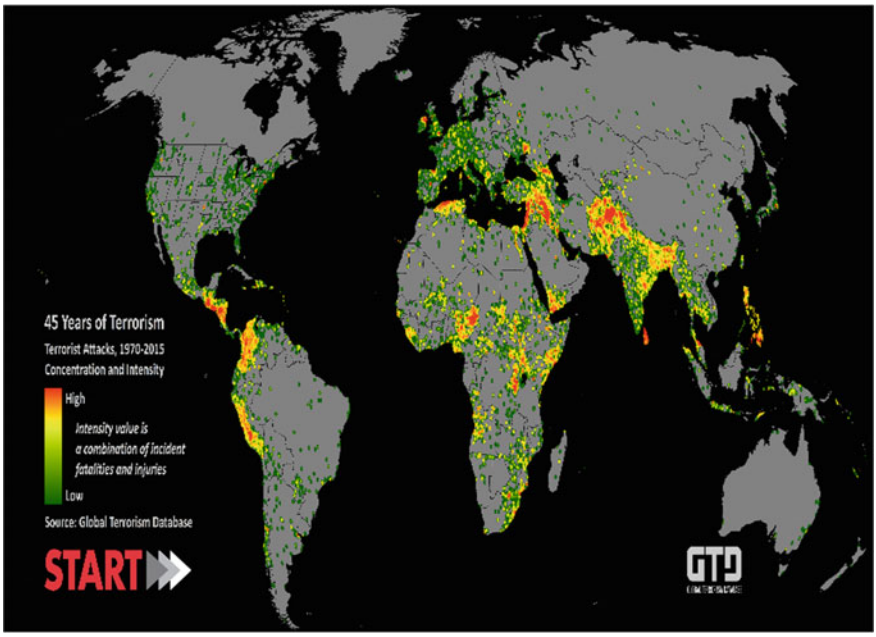


Fig. 1 GTD map for showing terror attacks worldwide. *Source* www.start.umd.edu

1.2 Graph and Neo4j

Neo4j is a JavaScript-based tool for creating and manipulating graphs. CQL or cipher query language is used for its operation [6]. The graph data science library in Neo4j may be used to apply various algorithms to the graph. Neo4j's computed embedding may be exported as a CSV file. The machine learning model will use the estimated embedding, degree, and centrality as significant features [6].

2 Literature Survey

Neo4j is a javascript-based tool used for creating and manipulating graph databases that use the Cypher query language, or CQL. The manipulation and mathematical operations on the graph database are made simpler by the preloaded plugins, such as the graph data science library (GDS) and awesome procedures on Cipher (APC). According to Felix Melchor Santos Lopez and Eulogio Guillermo Santos De La Cruz Neo4j gives the database atomicity, consistency, isolation, and durability (ACID), hence it is an excellent substitute for traditional SQL (Relational Database) [1]. For machine learning applications, Orange is considered as a data mining tool that serves as a substitute for the Python and R programming languages. With many

machine learning algorithms, including supervised learning techniques and unsupervised learning approaches, Orange tool was most recently released in 2016 by including a huge library for data preprocessing along with the utilization of data imputation block for removing null values from the dataset, PCA (Principle component analysis) block is commonly used for performing dimension reduction, wherein the data preprocessor block is used for data scaling, and support all the preprocessing algorithms. To show the machine learning use of Orange data mining tool, Musa Peker, Osman Ozkaraka, and Ali Sasar implemented five machine learning models on a diabetic dataset obtained from Dalaman State Hospital of Turkey [2]. Today's market offers a variety of data mining technologies, including R programming language, Rapid Miner, WEKA, Orange, and Kinme. Rapid Miner is language agnostic, whereas orange was created using C, C++ , Cython, and Python. Orange offers more freedom to the developers by offering them a load model block so that they may create their own models and send them to the orange tool [3]. The benefits and downsides of various data mining technologies were thoroughly compared by Ranjan et al. [3]. The end-user can implement a variety of machine learning models provided by WEKA using the java programming language. With a data training percentage of 66% and test data percentage of 34%, Ghada M. Tolan attempted to use machine learning models by including Naive Bayes, K-nearest neighbour, C4.5, ID3, and support vector machine. The dataset majorly used for terror attack classification is the global terror attack database, which is a copyright of the University of Maryland. It consists of 136 attributes and two lakh entries of incidents from the year 1970 to 2015. WEKA has both machine learning and deep learning support and is an open-source platform issued under GNU general public license [7]. Two of the most popular algorithms for classifying terror attacks are the decision tree and random forest algorithms. While the decision tree has never demonstrated accuracy above 75%, random forest algorithm with modified hyper-parameters has consistently demonstrated results above 90% [8]. Although GDBMS are now more widely accepted by data analysts, they nevertheless have their own drawbacks, such as high computer power requirements, longer calculation times, and more complicated algorithms when dataset sizes grow. Due to the large number of libraries that are filled with graph data science and their connection with the python programming language, Neo4j, Orient DB, and Titan are considered as the most promising graph database management technologies [9]. Neo4j operates twenty times more efficiently than conventional RDBMS, such as Postgre, when compared to the two types of RDBMS. Both the relational database and the graph database have their own advantages and thus it is impossible to say which is quicker because it relies entirely on the application for which it is being used [6]. More than twenty graph database solutions are now available on the market, including Orient DB, Arango DB White DB, Graph DB, Azure Cosmos DB, Fauna DB, Tiger Graph, Neo4j, Velocity DB, Memgraph, Titan, and many others. Of these, Neo4j and Tiger Graph are the two that perform the best. These graph databases are frequently used in the field of biomedical engineering to record patient names, identification numbers, diagnosis, and treatment information.

The advantages of graph databases over relational databases have also been demonstrated with a thorough comparison of the graph database frameworks by Timon-Reina et al. [10]. The graph database has a variety of uses, including network administration, social connectivity, biology, and the identification of fraudulent conduct. Compared to relational databases, it offers the developer more performance, flexibility, and agility [11]. Hybrid models and ensemble machine learning techniques also produce promising outcomes, with these techniques achieving results ranging from 87 to 97%. ROC, AUC, precision, recall, and F1 score are the performance measures used in the result analysis. This is mostly based on ROC curve analysis for each model [4]. Neural Network is another machine learning algorithm that when trained and tested for twenty epochs was able to give a mean squared error (MSE) of 0.180 by Ghada and Abou-El-Enien. Metaheuristic Optimization algorithms, which help in increasing the prediction accuracy of machine learning algorithms [5]. The GTD codebook gives an overview of the data collection methodology for the global terrorism database. The code is maintained by the University of Maryland as well as it is copyright of the same. The database can be used on an individual basis for study purposes and is provided by the admin on a request basis [12].

3 Methodology

3.1 Graph Creation

An application called Neo4j Sandbox was used to create the graphs. The global terrorism database was utilized as a source for building the graph and was imported into Neo4j by using the LOAD CSV cipher command. Using the CREATE command, several nodes were added to the graphs. Event Timing includes the date, time, and year of the occurrence; Event location includes the neighborhood, location, longitude, latitude, and Specificity, and attack types attacktype1, attacktype2, attack subtype 1, and attack subtype 2 were provided in the event info. Target type was composed of target types 1, 2, and target subtypes 1, and 2. Weapon type 1, weapon subtype 1, weapon type 2, weapon subtype 2, gun name, and gun type were all contained in the weapon type. Property damage, causality, and Ransome type were all the factors in causation. These nodes are all linked together through relationships. A cipher query language was used to generate and modify the graph. A subgraph of two nodes was constructed from the generated graph in order to compute graph embedding [7] (Fig. 2).

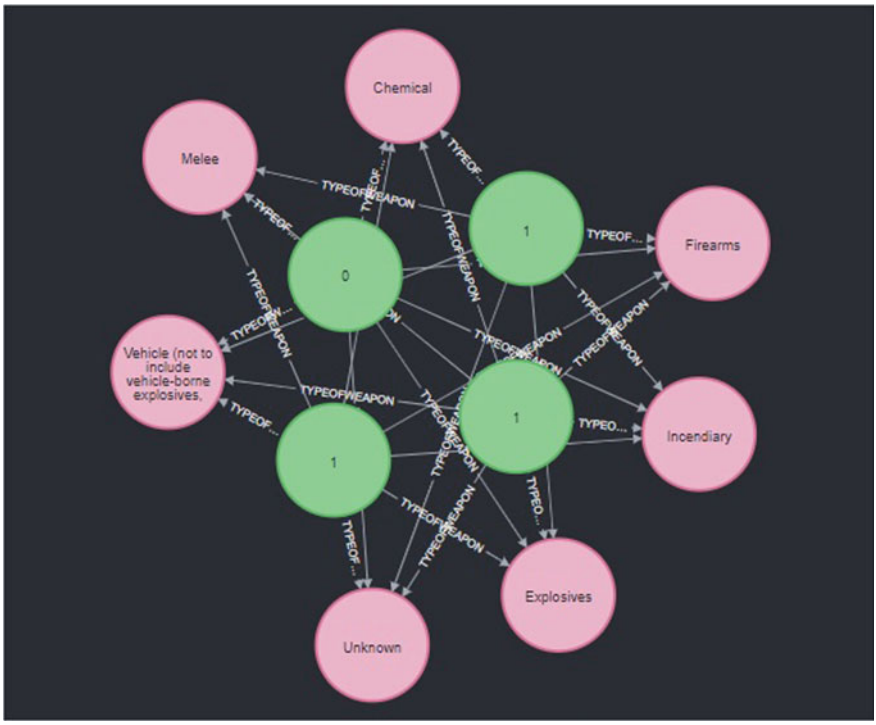


Fig. 2 Sample graph created using Neo4j Sandbox

3.2 Graph Embedding Calculation

Graph a higher dimension data is essentially represented in a lower dimension via embedding. They take the shape of a vector. In our example, embedding was calculated by using the node2vec technique. Node2vec algorithms operate based on random walks in the network. The graph is effectively represented in a lower dimension with a graph embedding by assuming a vector form [10]. The node2vec method was used to calculate embedding in our case. Node2vec methods use random network walks and are largely based on word2vec techniques. Fast random projection, node2vec, and Graphsage are the three techniques offered by Neo4j to compute node embeddings, node2vec is the approach used here. With the use of second-order random walks, the node2vec method creates a list of node identities that, when put together, constitute a sentence. This corpus of sentences is then used to calculate embedding vectors, also known as node embeddings or graph embeddings. Based on random walks, the node2vec method alternates between depth-first search and breadth-first search [11]. Although up to 10 embedding dimensions have been generated in the research, the embeddings between two nodes in a network will be calculated based on the Neo4j platform up to n dimensions.

4 Model Building

The orange data mining tool was used to create many machine learning models. The global terrorism dataset and computed embedding are the first two input datasets that are entered into the orange tool by utilizing the CSV file import block. A Data table block may be used to visualize the CSV's contents [2]. A Data table block may be used to visualize the CSV's contents. The data table block's output was provided as an input to the merge data block, which integrated the separate datasets into one dataset [3]. There are many null values in the merged dataset that can't be directly given as input into the models and thus an imputation block was utilized to eliminate those null values. The select column block was incorporated into the models once null values were eliminated. The selected column block is used to choose and remove characteristics from the dataset as well as to choose and configure the model's target variable. The purge domain block in the orange tool was used to delete and eliminate the redundant characteristics from the dataset since the data frame contains certain redundant attributes that were making the model's prediction accuracy redundant. Figure 3 shows the employed model in the orange tool where X resembles the name of the ML models.

Before feeding the dataset to the model, the dataset was scaled by using a data processor block. Overfitting is a severe problem that affects machine learning models most of the time. Here, principal component analysis (PCA) is used as a dimension reduction approach to solve this problem. The input data was then divided into

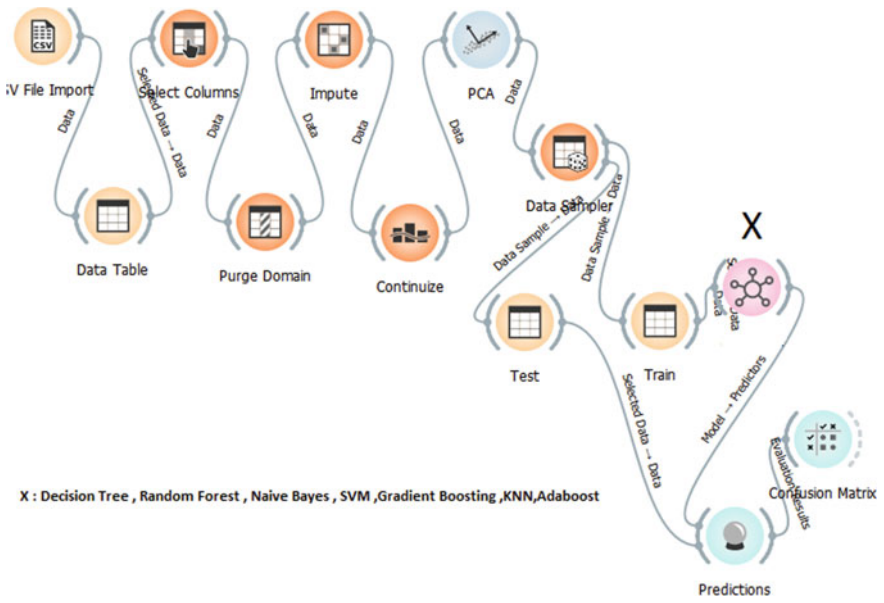


Fig. 3 Applied model in orange tool

training and testing data by using an 8:2 sampling ratio. Eighty percent of the data were used to train the model, and twenty percent were used to test it, according to the sampling ratio of 8:2. The model block was the next to be added, and it received its input from the training database and its output from that block was provided as input to the test and score block. The test data table serves as the second input for the test and score block, which also assess how well the machine learning model performed. Multiple building elements, such as a bar plot, line plot, heat map, and others, can be used to visualize the results. In our instance, a confusion matrix was employed to visualize the outcome.

4.1 Counting Null Values

The orange data mining tool's impute block was used to count and eliminate null values from the dataset. There is a choice to use the average and most frequent imputation algorithm, random value imputation algorithm, model-based imputer method, or fixed value or numeric value imputation algorithm. In this study, the most frequent and average imputation procedure was used to remove null values from the dataset. The data imputation block for removing null values from the dataset is shown in Fig. 4.

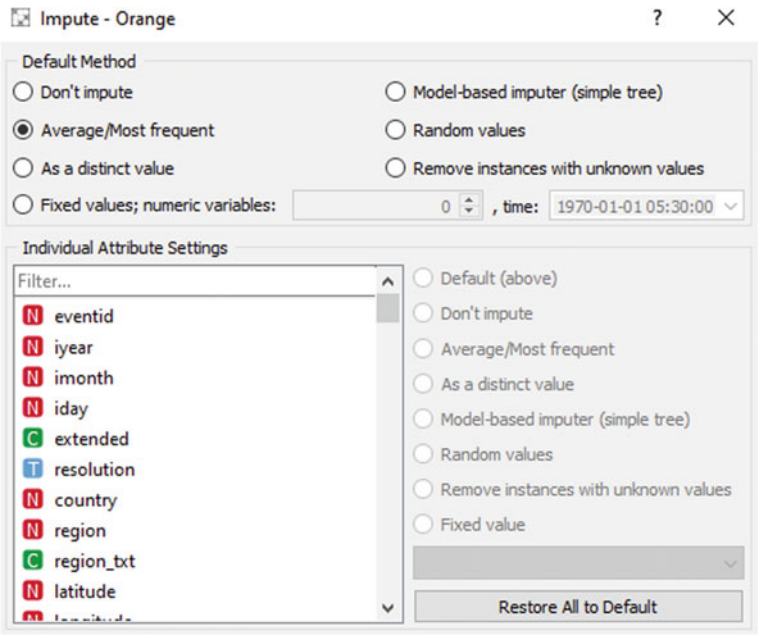


Fig. 4 Data imputation block in orange

4.2 Removal of Redundant Data

The purge domain block in the orange data mining tool is used to eliminate or discard redundant characteristics from the dataset. Three alternatives are provided by the purge domain block to eliminate redundant characteristics from the dataset: features, classes, and meta attributes. The three major functions performed by purge domain block are sorting, reducing, and removing features. The purge domain block for removing redundant attributes from the dataset is shown in Fig. 5.

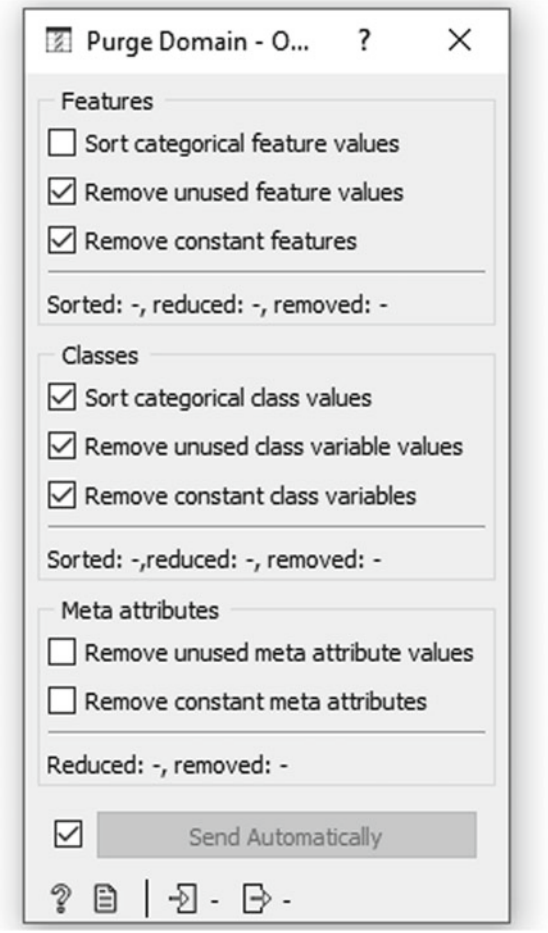


Fig. 5 Purge domain block in orange tool

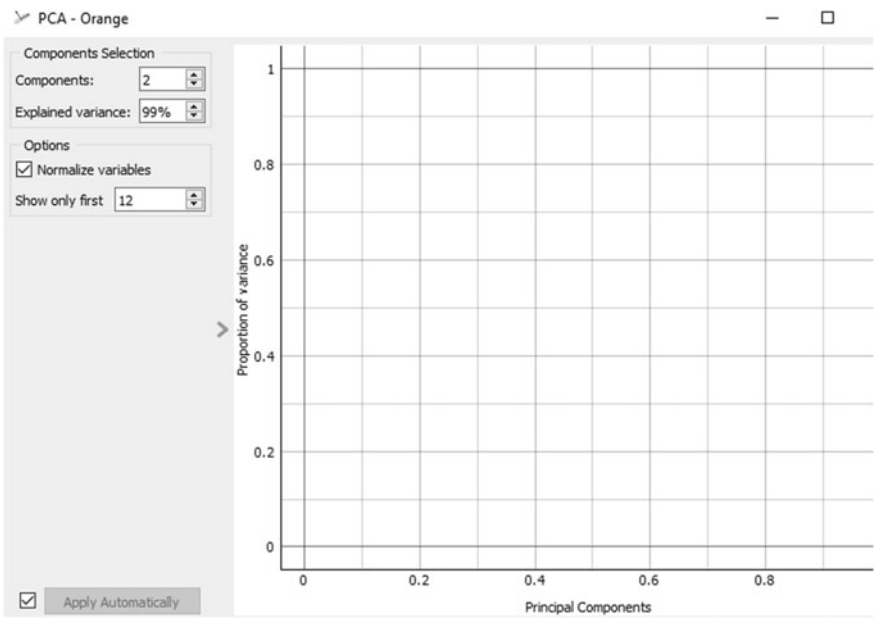


Fig. 6 PCA in orange tool

4.3 Dimension Reduction

Overfitting, which happens as a result of the dataset's many characteristics, is one of the main issues that machine learning models encounter. There are several methods for reducing the number of dimensions, including PCA (Principal Component Analysis) and LDA (Linear Discriminant Analysis). PCA (Principal Component Analysis) is used in this study to reduce the dimensions to two components. With a variance of 99%, the PCA block in the orange data mining tool is utilized for performing dimension reduction. The PCA block for dimension reduction from the dataset is shown in Fig. 6.

4.4 Data Scaling

The process of bringing the data into a certain range so the model can quickly learn and categorize is known as data scaling. The orange tool has a number of scaling techniques, including conventional scaling and center scaling. For data scaling, utilize the orange data processor block. Data preparation options available in the data preprocessor block include discretization, continuization, imputation, normalizing, randomization, and principal component analysis. The data processor block for data scaling from the dataset is shown in Fig. 7.

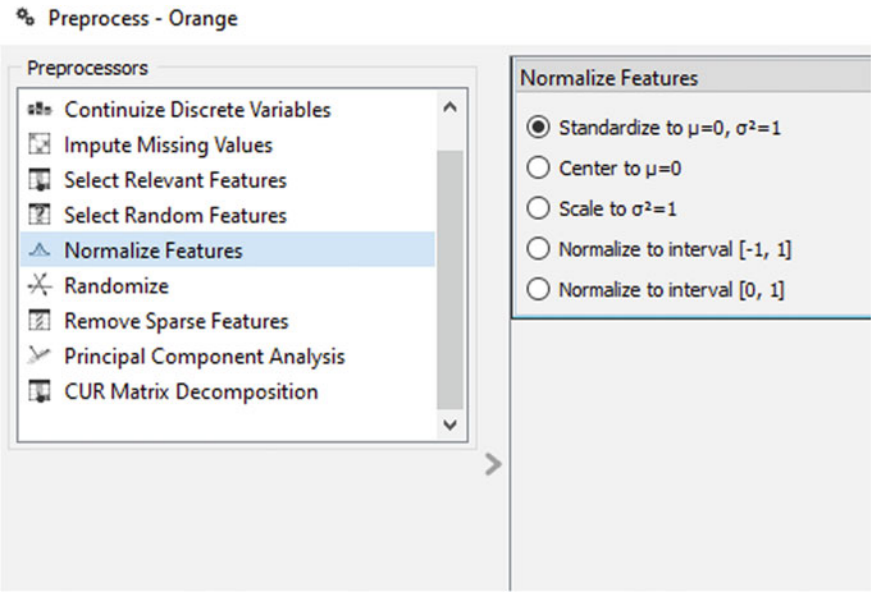


Fig. 7 Preprocess block in orange tool

5 Results and Analysis

The obtained research findings indicate that Random Forest (RF) with accuracy scores of 0.938, F1 scores of 0.920, precision scores of 0.936, and recall scores of 0.932 is the model that performs the best. SVM is the model that performs the poorest, with accuracy scores of 0.554, F1 scores of 0.830, precision scores of 0.850, and recall scores of 0.880. AUC, precision, recall, and F1 Score are the study's performance indicators. Table 1 shows the results obtained from the proposed research.

Table 1 Results obtained from the proposed research

Model	AUC	F1 score	Precision	Recall
RF	0.938	0.920	0.936	0.932
GB	0.931	0.955	0.954	0.955
KNN	0.743	0.850	0.842	0.870
Tree	0.572	0.903	0.902	0.911
SVM	0.554	0.830	0.850	0.880
NB	0.833	0.710	0.869	0.651
Adaboost	0.862	0.927	0.931	0.925

To determine accuracy, apply the formula below:

$$Accuracy = (TP + TN) \Big/ (TP + TN + FP + FN)$$

To determine the precision, apply the formula below:

$$Precision = TP \Big/ (TP + FP)$$

To determine the recall, apply the formula below:

$$Recall = TP \Big/ (TP + FN)$$

To determine the F1 Score, apply the formula below:

$$F1\,Score = 2 * (Precision * Recall) \Big/ Precision + Recall$$

where TP is the true positive classified sample by the model, TN is the true negative classified sample by the model, and FN is the false positive classified sample by the model, FN is the false negative classified sample by the model. Figure 8 shows the confusion matrix for the random forest model.

Figure 8 shows the confusion matrix for the random forest model.
Figure 9 shows the confusion matrix for the gradient boosting model.
Figure 10 shows the confusion matrix for the KNN model.
Figure 11 shows the confusion matrix for the decision tree model.
Figure 12 shows the tree diagram for the classification model.
Figure 13 shows the confusion matrix for the SVM model.
Figure 14 shows the confusion matrix for the naive bayes model.
Figure 15 shows the confusion matrix for the Adaboost model.

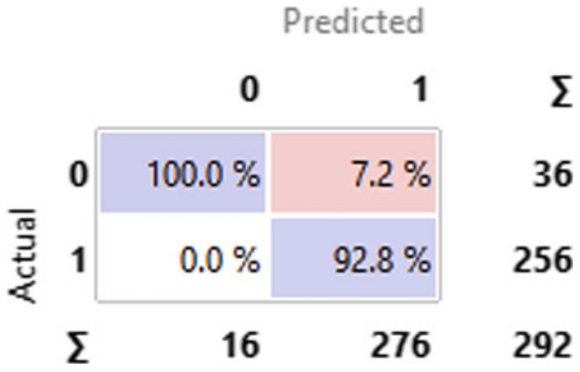


Fig. 8 Confusion matrix for random forest

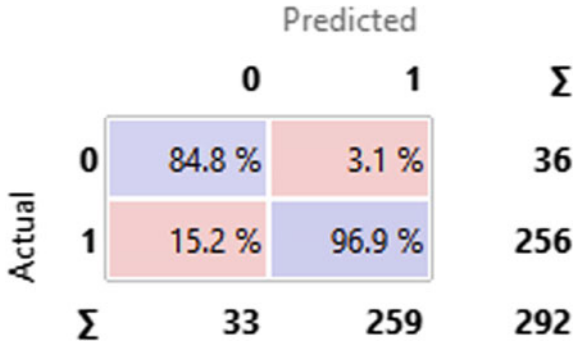


Fig. 9 Confusion matrix for gradient boosting

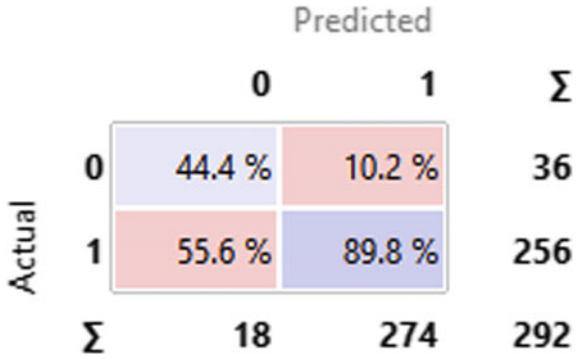


Fig. 10 Confusion matrix for KNN

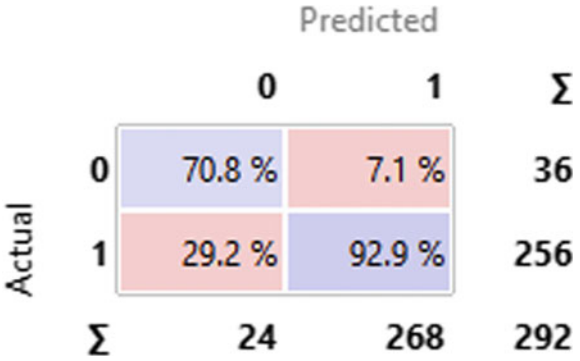


Fig. 11 Confusion matrix for decision tree

6 Conclusion

Tools like Orange and WEKA can be very useful in the absence of knowledge on python programming language. Graph features such as graph embedding can act as useful features in the classification process. Random Forest (RF) and Gradient

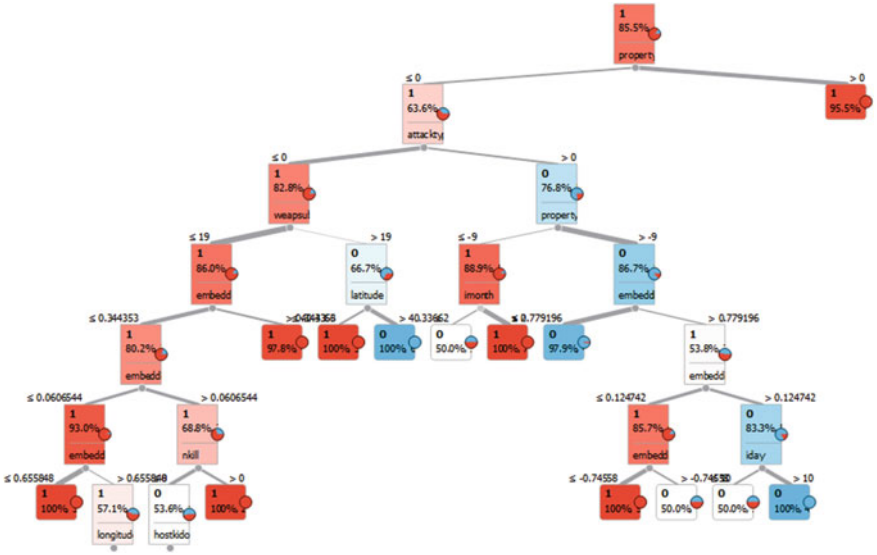


Fig. 12 Tree diagram for decision tree

Fig. 13 Confusion matrix for SVM

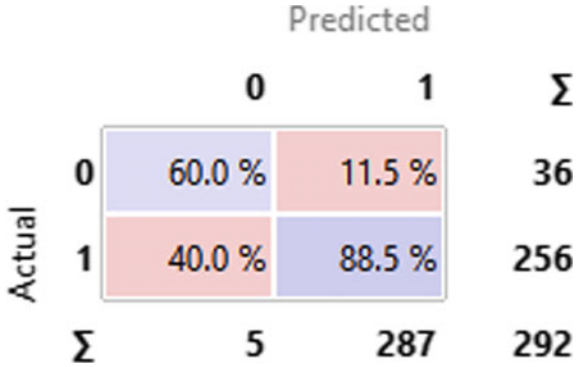
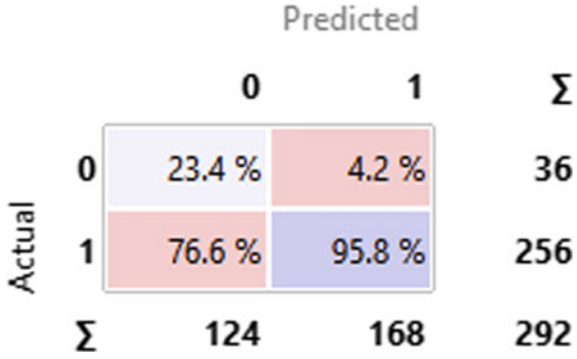


Fig. 14 Confusion matrix for naive bayes



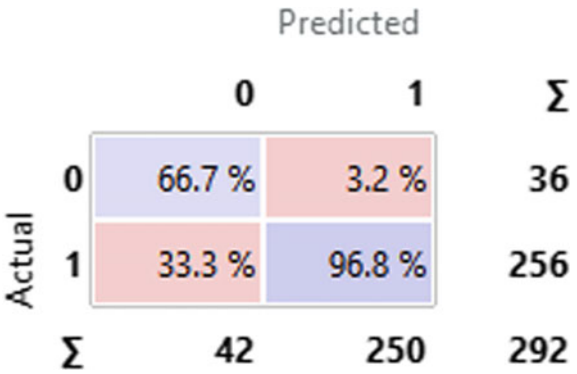


Fig. 15 Confusion matrix for Adaboost

Boosting (GB) techniques are the most promising techniques used for the classification of terror attacks by using the orange data mining tool. According to our findings, Random Forest (RF) can provide an accuracy score of 0.938, F1 score of 0.920, precision of 0.936, and recall of 0.932 on a training and testing ratio of 8:2. The worst performing model was SVM, which gave an accuracy score of 0.554, F1 score of 0.830, precision of 0.850, and recall of 0.880. Although orange is a very diverse tool with its own limitation including less flexibility, a predefined and limited number of algorithms, and less customization of blocks. Graph embedding has compensated the inconsistency in the dataset and improves the prediction accuracy of the model.

References

1. Lopez FMS, De La Cruz EGS (2015) Literature review about Neo4j graph database as a feasible alternative for replacing RDBMS. Revista de la Facultad de Ingenieria Industrial 1560–9146
2. Peker M, Özkaraca O, Şaşar A (2018) Use of orange data mining toolbox for data analysis in clinical decision making: the diagnosis of diabetes disease. In: Expert system techniques in biomedical science practice
3. Ranjan R, Agarwal S, Venkatesan S (2017) Detailed analysis of data mining tools. Int J Eng Res Technol (IJERT) 2278–0181
4. Python A, Bender A, Nandi AK, Hancock PA, Arambepola R, Brandsch J, Lucas TCD (2021) Predicting non-state terrorism worldwide. Sci Adv
5. Soliman GMA, Abou-El-Enien THM (2019) Terrorism prediction using artificial neural network. Revue d'Intelligence Artificielle
6. Macák M, Stovcik M, Buhnova B (2020) The suitability of graph databases for big data analysis: a benchmark. IoTBDS
7. Tolan GM, Soliman OS (2015) An experimental study of classification algorithms for terrorism prediction. Int J Knowl Eng
8. Huamaní EL, Alicia AM, Roman-Gonzalez A (2020) Machine learning techniques to visualize and predict terrorist attacks worldwide using the global terrorism database. Int J Adv Comput Sci Appl (IJACSA)

9. Pokorný J (2015) Graph databases: their power and limitations. In: Computer ınformation systems and ındustrial management
10. Timon-Reina´ S, Rincon M, Martínez-Tomas R (2021) An overview of graph databases and their applications in the biomedical domain. Database J Biol Database Curation
11. ShefaliPatil G, Bhatia A (2014) Graph databases—an overview. Int J Comput Sci Inf Technol
12. Global terroism database. https://www.start.umd.edu/gtd/downloads/Codebook.pdf