Social Network Analysis and Mining (2024) 14:228
https://doi.org/10.1007/s13278-024-01389-5

ORIGINAL ARTICLE

Deepthreatexplainer: a united explainable predictor for threat comments identification on Twitter

Anna Nazarova[1] · Muhammad Shahid Iqbal Malik[1,4] · Dmitry I. Ignatov[1] · Ibrar Hussain[2,3]

Received: 24 July 2024 / Revised: 4 September 2024 / Accepted: 16 November 2024

Abstract
Identification of threatening comments on social media platforms has recently gained attention. Prior approaches have addressed this task in some low-resource languages but the interpretability of results was not studied. In addition, approaches in the English language are minimal. To support explainable predictive inference, this research proposes an inherently explainable model for threat comment identification on Twitter. The proposed system incorporates the strengths of Bayesian logistic regression with optimal variational capacity and facilitates the estimation of salient features. Furthermore, the Optimal Variational-Bayesian Logistic Regression (OVB-LR) model can handle the limited labeled dataset issue, achieving the highest performance in classification. The proposed framework automatically mines the threat-related context in language and provides intrinsic explainability for its prediction. This is achieved by posterior-probability approximation, and feature weight calculation to select salient features. For evaluation, a new dataset containing English tweets is designed for threat comment identification. The performance of the proposed framework is evaluated on the threat dataset, and compared with four classical Machine Learning (ML) models (logistic regression, random forest, support vector machine, and k-nearest neighbors) using two feature extraction methods: ELMo embeddings and word uni-gram. The results exhibit that the proposed framework achieves benchmark performance and outperforms four ML models, achieving 81.25% accuracy, 80.85% f1-score for threat class, and 81.24% macro f1-score stably on the newly designed dataset. Furthermore, the OVB-LR model demonstrates comparable interpretations and selects important features that align with features inferred by two post-hoc: Shapley Additive Explanations (SHAP) and Accelerated Model-agnostic Explanations (AcME) Explainable Artificial Intelligence methods. The findings have practical implications for commercial applications and future research.

Keywords Threatening text · Inherently explainable · Twitter · ELMo · Word unigram · Machine learning

1 Introduction

Social media platforms like Twitter, Facebook, and YouTube have become an essential part of our daily lives. Statistics show that over 62.3% of the population uses social networks (Alda 2021). This generates a large amount of information, ranging from personal experience to verified news, which is being spread quickly. Social networks allow for anonymity to online users when they want to share comments and/or posts. This facilitation can reduce self-control and allow the expression of negative content (Hoang and Pishva 2014; Liu et al. 2210), including threats. Such posts can make readers uncomfortable and may lead to real-life crimes. According to the Cambridge Dictionary, a threat is defined as; 'a suggestion that is something unpleasant or violent, especially if a necessary action/step is not taken.' Threatening others may involve the use of offensive, derogatory, or discriminatory language based on the factors of race, gender, nationality, religion, ideology, interests, etc. The target of threat is not only an individual but also an entire group and it can be achieved through various means, such as textual information, provocative images, videos, or soundtracks.

✉ Muhammad Shahid Iqbal Malik
mumalik@hse.ru; shahid.msimalik@gmail.com

1 Department of Computer Science, National Research University Higher School of Economics, 11 Pokrovskiy Boulevard, Moscow, Russian Federation 109028

2 Department of Software Engineering, University of Lahore, Lahore 54000, Pakistan

3 Faculty of Engineering and Information Technology, Shinawatra University, Bangtoey Samkhok, Pathum Thani 12160, Thailand

4 Department of Computer Science, HITEC University Museum Road, Taxila 47080, Pakistan

Published online: 03 December 2024

Due to the abundance of information available online, it is challenging to filter out harmful content. Social media platforms have been used to spread threats and amplify the dissemination of such information. For example, terrorist organizations have been observed to communicate with each other, spread propaganda, and recruit perpetrators using various social media platforms (Hossain 2015). To prevent or at least reduce the flow of such information, various methods are being considered to regulate the dissemination of threatening content by employing various forms of moderation, including manual moderation by humans and automatic moderation using trained models. However, both manual and automatic moderation face several hurdles and challenges. Manual moderation cannot filter the entire flow of information, whereas automatic moderation requires careful preparation before use. In addition, the accurate identification of hateful, offensive, or threatening texts relies primarily on the data used to train the model. If data is not carefully prepared for large-scale use, the process of using it can result in large numbers of false positives and false negatives, because:

1. In literature, several definitions are being used for equivalent concepts, which results in making most of the available corpora incompatible.
2. The words that are essential in the classification of text may not be popular, important, or maybe outdated.
3. The amount of data may not be sufficient.

This can lead to user outrage and may not effectively prevent the spread of harmful content. The study (Fortuna et al. 2020) analyzed hate speech datasets and concluded that although they have the same objectives, they have concentrated examples of specific categories of hate speech, which differ from definitions used in other studies.

Having characteristics is insufficient to ensure good model performance. Additionally, it is crucial to understand the logic behind a model's predictions. To achieve this, a range of XAI models are available. XAI has gained significant attention due to the increasing demand for the interpretability of ML and DL models across various fields (Rudin et al. 2022). The primary objective of XAI is to develop models that should be interpretable to humans, and provide meaningful inferences that can be understood and trusted. Considering types of XAI models, the first type uses ‘white-box’ or ‘glass-box’ models, which are simple machine learning models and can be understood without a need for additional models. The second type of XAI uses 'grey-box' models, which can be interpreted to some extent if they are carefully designed. For the third type of XAI, separate XAI models are needed to explain the results of existing ‘black-box’ models, which are problematic to trust and understand due to their complex architecture (Saarela and Jauhiainen 2021). It is important to note that there is no scientific evidence for a general trade-off between accuracy and interpretability. Many ML models demand certain constraints to improve interpretability, which can limit the maximum achievable accuracy. However, with careful design, a good balance between interpretability and accuracy can be achieved (Du et al. 2019).

This study addressed two challenges (interpretability and performance) simultaneously and proposed an interpretable and robust threat comment identification model that achieved benchmark performance. The proposed framework utilized the strengths of intrinsic explainable model. Specifically, the potential of Bayesian logistic regression with optimal variational capability (Liu et al. 2024) is adopted for the design of the proposed OVB-LR model. To the best of our knowledge, it is the first attempt to design an interpretable threatening comment identification model with state-of-the-art performance. The OVB-LR model's classification performance is compared with classical ML models, and its interpretability is compared with standard SHAP and AcME interpretable approaches. Furthermore, the OVB-LR model explored the relationship between salient features and intrinsic explainability and highlighted the important features. The contributions of this study are presented below:

1. This article proposed an intrinsic explainable model with salient features inference for detecting threatening comments on Twitter with benchmark performance.
2. The prior dataset has annotation issues. Therefore, a new corpus for identifying threatening comments in English tweets has been developed.
3. The proposed framework provides valuable insights by highlighting important features, thus offering a fresh perspective on explainable ML for threat comment detection.
4. The OVB-LR model presented benchmark performance by achieving 81.25% accuracy, and 81.24% macro f1-score, and outperformed the classical baselines.
5. The proposed framework’s interpretability is comparable and inference of important features is aligned with the outcomes of SHAP and AcME XAI models.

The paper is structured as follows: Sect. 2 provides a literature review of recent studies for offensive, threatening, and extremist text identification and XAI models. Section 3 presents the proposed methodology and dataset construction in detail. Section 4 describes the details of the proposed model’s parameters and baselines for the experimental setup. The results are presented in Sect. 5 with analysis by comparing the proposed model and baselines. Section 6 presents the discussion and limitations of the study. Finally, Sect. 7 provides the conclusion and discusses future directions. The list of abbreviations is added in Table 1.

Table 1 Glossary of key terms

ξ	Variational parameters, which were introduced to approximate posterior probability distribution	Σ_N	Covariance matrix of the approximate Gaussian distribution
μ_N	Mean vector of the approximate Gaussian distribution	w	Optimal regression coefficients
D	Number of features in dataset	N	Number of samples in dataset
$\phi(.)$	Basis function that transforms the original feature $x \in R^D$ to a specific feature space	α	Precision parameter of the prior distribution
AcME	Accelerated Model-agnostic Explanations	ANN	Artificial Neural Network
ARB	Arabic language	BERT	Bidirectional Encoder Representations from Transformers
BiLSTM	Bidirectional Long Short-Term Memory	BOW	Bag-of-words
CBiLSTM	Convolutional Bidirectional Long Short-Term Memory	CI	Confidence Interval
CNN	Convolutional Neural Network	DL	Deep Learning
DNN	Deep Neural Network	DT	Decision Tree
ELMo	Embeddings From Language Model	ENG	English
GB	Gradient Boosting	GloVe	Global Vectors for Word Representation
KNN	K-nearest Neighbors	LIME	Local Interpretable Model-agnostic Explanations
LR	Logistic Regression	LSA	Latent Semantic Analysis
LSTM	Long Short-Term Memory	ML	Machine Learning
MLP	Multilayer Perceptron	NN	Neural Network
OVB-LR	Optimal Variational Bayesian- Logistic Regression	PCA	Principal Component Analysis
RF	Random Forests	RL	Reinforcement Learning
RL	Reinforcement Learning	SHAP	Shapley Additive Explanations
VI	Variational Inference	URD	Urdu language
TF-IDF	TF—term frequency, IDF—inverse document frequency	XAI	Explainable Artificial Intelligence
SGD	Stochastic Gradient Descent	SVM	Support Vector Machine
AB	AdaBoost	RAE	Radicalization and Extremism
LDA	Latent Dirichlet allocation		

2 Related works

This section briefly reviews the research works done so far for abusive, threat, and extremism identification in social media as well as a summary of XAI approaches and advancement in this domain.

2.1 Offensive, threatening, and extremist views identification

A threat takes many forms, from verbal to the use of mass media. For a threat to be considered successful, it must involve at least two participants: the speaker (who intends to cause fear or alarm), and the listener (who accurately infers the speaker's intentions). Threatening is a complex concept and is difficult to define objectively. Although laws in some countries have attempted to establish a basic definition of threat, it is important to recognize that the listeners' experience and attitude may influence their perception of another's actions. Instances of mass threats can be seen in various parts of the world. For example, in 1994, the media incited violence that led to genocide in Rwanda (Viljoen 2005). Similarly, during the post-election violence in Kenya in 2007–2008, some Kenyan media outlets, particularly local indigenous radio stations, were accused of spreading hate messages and inciting ethnic hatred by media monitors, human rights groups, politicians, and journalists (Somerville 2011).

Studies that examine threat content also consider extremist and radical content, such as the detection of jihadism. In addition to binary categorization tasks that identify the presence or absence of extremist/radical content, attempts have been made to determine which specific extremist group the texts belong to Scanlon and Gerber (2015). They utilized LDA for the analysis of content and showed that LDA-based topics are influential predictors compared to baselines. Then an article (Kaati et al. 2015) conducted a study to identify tweeps ("supporters of jihadist groups") using data-dependent (word bi-grams, etc.) and data-independent (stylistic, emotions, etc) features. They achieved the best performance (99.51%) with the AdaBoost classifier and data-dependent features on the English dataset but did not address the explainability of prediction.

Experiments have been conducted using various meta-data features such as user profile (the number of followers, friends, and tweets), location, content data, and Twitter handle data (length of the handle, and sentiment of the handle). A study (Alvari et al. 2019) proposed a detection mechanism for identifying extremist users. Various user features are explored and their framework showed effective performance. Likewise, another study handled the same task and proposed a detection system for extremist users, and content adopters (Ferrara 2016). They used metadata, network, and temporal features and achieved an 87.4% f1-score. In addition, prior studies have explored the use of sentiment, lexicon, stylometric, as well as time pattern features for terrorism identification (Azizan and Aziz 2017), and radical social media content detection (Gupta et al. 2017). Using naïve bayes and AdaBoost ML models, they obtained the highest performance.

Addressing radical and extreme behavior identification, the study (Nouh et al. 2019) proposed a system to identify radical views from social media content. They explored psychological, behavioral, and linguistic features with several ML models. The best performance is obtained with linguistic and psychological features in combination with the RF model. Similarly, Sharif et al. (Sharif et al. 2019) developed a framework for extremist behavior detection on Twitter. They demonstrated that the quality of results can be preserved by using PCA to reduce feature dimensions. The SVM model with word bi-gram features achieved the best results (84.71% accuracy). Later, another work (Mussiraliyeva et al. 2020) derived a detection mechanism for radicalism and extremism in the Kazakh language. Various ML models were explored and they obtained 89% performance with the combination of the GB model and word2vec embeddings. Likewise, Arabic tweets are categorized into extremist or non-extremist by exploring various ML and DL techniques (Aldera et al. 2021). The fine-tuned BERT model outperformed all other models and achieved 97.49% accuracy.

Regarding threat detection, a violent threat identification model for YouTube comments was developed by Ashraf et al. (2020). The BOW, TF-IDF, GloVe, and FastText embeddings are explored with DL models. The best results are achieved with the TF-IDF and BiLSTM models. However, they did not address the task of explainability of results. The study (Amjad et al. 2021) developed a dataset to classify threat instances in Urdu and determine whether they are directed toward a group or an individual. They utilized various machine learning and deep learning models and the MLP classifier with the word n-gram features outperformed in detecting threat content (72.50% in accuracy) and SVM with fastText features obtained the best results for the target identification task (75.31% accuracy). Another work (Hussain et al. 2022) developed a model for the detection of offensive content on Facebook posts. They explored various ML and feature extraction methods such as n-grams, TF-IDF, BOW, and word2vec. An ensemble model with the combination of BOW + TF-IDF + word2vec as features achieved the best accuracy of 89.23%.

Later, the identification of threat views and their targets in Urdu is proposed by the study (Malik et al. 2023a). They explored fine-tuning Urdu-BERT along with various ML and DL models and feature extraction methods, including LSA. Their experiments revealed that fine-tuned Urdu-BERT achieved the best performance with 87.5% accuracy. Another study developed a multi-lingual threat comment identification framework for English and Urdu languages (Rehan et al. 2023). The proposed model is based on fine-tuning MuRIL and Urdu-RoBERTa and achieved benchmark performance but did not address the interpretability of the prediction task. Recently, violence incitation comments have been handled by a study (Khan et al. 2024), that proposed an identification system for the Urdu language. They developed a new dataset for violence incitation detection and conducted experiments using traditional ML and DL models. The 1D-CNN with word unigram model showed a benchmark performance by demonstrating 89.84% accuracy. However, all these studies missed the explainability of their results, thus leaving the hidden logic as a block box. Generally, all studies use similar machine learning and deep learning models. However, each study attempted to use a unique method of feature extraction or a different dataset. The summary of related studies is presented in Table 2.

2.2 Explainable machine learning approaches

Most classification models are often referred to as 'black boxes' due to the ambiguity of their decision-making process. Understanding the causes of a model's outcomes is crucial, regardless of its domain of application. Knowing how and why a model makes certain decisions can:

1. Increase the developer's confidence in its correctness.
2. Provide more informative answers.
3. Boost consumer and business confidence in the model's results.
4. Ensure that the model's results comply with laws.

To achieve explainable results, XAI techniques need to be utilized. The primary objective of XAI is to generate models that are interpretable by humans and produce meaningful and trustworthy results. A model is deemed trustworthy based on various criteria, including robustness, interpretability, explainability, fairness, interactivity, and stability. In general, explainable models are categorized (Ali et al. 2023) in the following ways:

1. **Family of inherently interpretable models**—These models are initially considered as explainable, i.e. white-box models. However, these models have a disadvan-

Table 2 Summary of related work for extremism, radicalism, threat, and offensive language detection in social media

Year [ref]	Task	Data source [languages]	Feature extraction	Supervised models
2015 (Kaati et al. 2015)	Extremism detection	Twitter 6279 samples [ENG, ARB]	Data-independent, & dependent features	AB
2016 (Ferrara 2016)	Extremism detection	Twitter 3 million samples [ENG]	User metadata & activity, Timing, Network statistics	LR, RF
2017 (Azizan and Aziz 2017)	RAD detection	Twitter 1480 samples [ENG]	Sentiment features, Lexicons features	NB
2017 (Gupta et al. 2017)	Radicalization detection	Twitter 48,644 samples [ENG]	Stylometric, Time Pattern features	RF, AB, NB, SVM
2019 (Alvari et al. 2019)	Extremism detection	Twitter 300 k samples [ENG]	Twitter handles, Profiles & content features	char-LSTM, SVM, NB, LR, AB, RF
2019 (Nouh et al. 2019)	RAD detection	Twitter 17 k samples [ENG]	Textual, psychological features, and Behavioral features	RF, SVM, KNN, NN
2019 (Sharif et al. 2019)	Extremism detection	Twitter 7500 samples [ENG]	TF-IDF, PCA	SVM, NB, DT, RF, KNN, Ensemble
2020 (Ashraf et al. 2020)	Threat detection	YouTube 1388 samples [ENG]	BOW, Glove, FastText, TF-IDF	CNN, LSTM, BiLSTM
2020 (Mussiraliyeva et al. 2020)	RAD detection	Vkontakte, 200 samples [KAZ]	TF-IDF, word2vec	GB, RF
2021 (Aldera et al. 2021)	Extremism detection	Twitter 89,816 samples [ARB]	N-grams, TF-IDF, word2vec	LR, SVM, NB, RF, BERT
2021 (Amjad et al. 2021)	Threat detection	Twitter 3564 samples [URD]	Word and char n-grams with TF-IDF, FastText	LR, AB, RF, SVM, MLP, CNN, LSTM
2022 (Hussain et al. 2022)	Offensive language detection	Facebook 7500 samples [URD]	N-grams, TF-IDF, BOW, word2vec	LR, SVM, RF, SGD, Ensemble model
2023 (Rehan et al. 2023)	Multi-lingual threat detection	Twitter [ENG, URD]	MuRIL, Urdu-RoBERTa,	Fine-tuning MuRIL and Urdu-RoBERTa
2023 (Malik et al. 2023a)	Threat and target detection	Twitter 2400 samples [URD]	N-grams, TF-IDF, BOW, FastText, BERT, LSA	LR, RF, SVM, KNN, NB, BERT
2024 (Khan et al. 2024)	Violence Incitation detection	Twitter 4804 samples [URD]	N-grams, TF-IDF, word2vec, fastText, Urdu-BERT	SVM, NB, AB, LR, RF, CNN, BiLSTM

tage—their metrics compared to black-box models, are much lower. Such models include classical ML methods such as logistic regression, decision trees, and k-nearest neighbors. In addition, researchers have upgraded some classical ML algorithms to improve their metrics and added explainability. Some examples of this family are the Super-sparse linear integer model (Ustun and Rudin 2016), rule-based approach (Jung et al. 1702), ANN-DT (Schmitz et al. 1999), interpretable decision sets (Lakkaraju et al. 2016), clustering-based approach (Saisubramanian et al. 2020), optimal Bayesian logistic regression (Liu et al. 2024). However, several researchers claimed that the tradeoff between performance and interpretability does not always hold (Rudin et al. 2022).

2. **Hybrid explainable models**—The models that incorporate an interpretable modeling technique alongside an advanced black-box method. These models have better metrics compared to inherently interpretable models. An example of such a model is the algorithm, which combines a k-nearest neighbors algorithm with a deep neural network (Papernot and McDaniel 1803). Another example is the model developed by Alvarez Melis and Jaakkola (2018), which generalized a linear classifier to improve the interpretability of results. The study (Al-Shedivat et al. 2020) introduced contextual explanation networks, which use probabilistic models for prediction and generate parameters for intermediate graphical models used for prediction and explanations. Then, another study (Brendel and Bethge 1904) developed a model to approximate CNNs and produce explainable results. Research has been conducted on the use of Boolean logic (Widmer et al. 2023), predicate —logic (Ciravegna et al. 2023), and first-order logic axioms (Jaeger 1403) for explanations. Additionally, interpretability results have been achieved for reinforcement learning tasks, such as improving existing machine learning-based scene graphs (Amodeo et al. 2022) and achieving interpretability using Myerson values (Angelotti and Díaz-Rodríguez

2023). The study (Bennetot et al. 2022) created a model that combines a DNN with a symbolic knowledge base. Furthermore, the work (Kaczmarek-Majer et al. 2022) successfully converted SHAP-generated model explanations into linguistic summaries.

3. **Joint prediction and explanation**—The models that are explicitly trained to explain their predictions. In this category, most of the models are designed to solve only one category of tasks (sound, image, text, etc.). One variant of this category is a model that was trained with a label containing an explanation and an output (Hind et al. 2019). Another example is a model that attempted to answer a question textually and indicate in the image which part influences the given answer (Park et al. 2018). For image data, a model classified the image by identifying prototypical parts and combining the data obtained from the prototypes for final classification (Chen 2019). Another work is the use of reinforcement learning to explain why the image was assigned to a specific class (Hendricks 2016). A special loss function can be used to give preference to certain parts of an object within a class category while remaining neutral towards images from other classes, a special loss function is designed (Zhang et al. 2018).
4. **Explainability through architectural adjustments**—The architectures of these models are modified so that some aspects of them (outcome, importance of parameters, etc) can be explainable. This can be achieved through various methods, including regularizing models that are difficult to simulate (Wu et al. 2018) and teaching the model what to focus on to avoid meaningless statistical errors in the data (Ghaeini et al. 1902).
5. **Post-hoc explanation—**Models that generate additional characteristics during training which can be used by additional algorithms to analyze already trained models. These additional algorithms are used to explain the results obtained by the model. The post-hoc models such as SHAP (Lundberg and Lee 2017), AcME (Dandolo et al. 2023), and LIME (Ribeiro et al. 2016) are well-known. Although there are many explainable models available, post-hoc models remain a convenient solution due to their standalone nature. However, post-hoc models have some drawbacks:

   (a) They can be computationally slow due to the additional parameters required for construction and analysis.
   (b) The explanations provided by these models are based on assumptions, which may not always guarantee the accuracy and truthfulness of the results.
   (c) It is possible to manipulate these models to produce desired outputs.

6. **Other methodologies—**These models are not included in any other categories. An example of this category is the model which aims to find optimal lists of rules to reduce the empirical risk of a given training data set, as described by the study (Angelino et al. 2018).

The summary details of XAI models can be found in Table 3.

2.3 Research gap

Previous research has primarily focused on detecting radicalization and extremism, with only recent studies address the identification of threatening comments. Arabic, Urdu, and other low-resource languages are used in this area due to the availability of the datasets for analysis, including those in the public domain. However, some authors have highlighted issues with data annotation. Also, the majority of datasets were created for other tasks. We found the following limitations in the literature:

1. There is no appropriate dataset available for the said task in the English language.
2. As XAI is increasingly popular for various tasks, there have been no experiments conducted to use XAI to interpret the results for identifying threatening content.
3. Prior research on intrinsic interpretability is limited especially for NLP tasks.

Therefore, it is necessary to address these issues and advance the field by designing affective intrinsic interpretability models for comprehensive explanations of threat comment identification.

3 Proposed methodology

To address the above-mentioned issues, a methodology is designed to classify threat comments on Twitter. Using this methodology, it is possible to get the interpretation of classification results by employing an inherently explainable model. The proposed framework (OVB-LR) is compared with state-of-the-art post-hoc explainable methods (SHAP and AcME) to evaluate the effectiveness of the interpretation of the proposed model. The classification performance of OVB-LR is also compared with four classical ML models. Furthermore, a new dataset for the English language is designed for the threat comments identification.

This section describes the architecture of the proposed methodology developed for the binary classification of threat comments and an appropriate explanation of prediction results. Figure 1 illustrates the proposed framework, which consists of several steps, including data preparation,

Table 3 Summary of XAI models utilized for various tasks

Year [Ref]	Tasks addressed	Description
Family of inherently interpretable models		
2015 (Ustun and Rudin 2016)	Medical scoring system (classification)	Based on Linear Regression
1999 (Schmitz et al. 1999)	Specie age prediction (regression)	Based on decision-tree
2017 (Jung et al. 1702)	UCI dataset (classification)	Rule-based approach
2016 (Lakkaraju et al. 2016)	Bail outcomes, student performance, and medical diagnosis	Based on decision sets
2020 (Saisubramanian et al. 2020)	Violent crimes, Adult and Traffic accident data	Clustering-based approach
2024 (Liu et al. 2024)	Medical diagnosis (classification)	Based on Logistic Regression
Hybrid explainable models		
2018 (Papernot and McDaniel 1803)	Digits, and traffic sign detection (classification)	Combination of KNN and DNN
2018 (Alvarez Melis and Jaakkola 2018)	Medical diagnosis, Specie age, and Ionosphere condition prediction (classification)	Generalize a linear classifier by learning its features, and associated coefficients
2019 (Brendel and Bethge 1904)	ImageNet (classification)	ResNet-50 variation
2020 (Al-Shedivat et al. 2020)	Poverty, movie review, digit classifications	Probabilistic models to generate parameters for intermediate graphical explanations
2023 (Ciravegna et al. 2023)	Medical diagnosis, digits detection, electoral democracy prediction (classification)	Logistic models with Boolean and predicate logic to explain the results
2023 (Widmer et al. 2023)	Google Open Images (object detection)	Based on Boolean logic to explain the results
2014 (Jaeger 1403)	Synthesize data (classification)	First-order logic axioms to explain results
2022 (Amodeo et al. 2022)	Predicate, and visual phrase detections	Reinforcement learning technique
2022 (Bennetot et al. 2022)	Architectural style classification, Object detection	DNN with a symbolic Knowledge Base
2022 (Kaczmarek-Majer et al. 2022)	Medical diagnosis (classification)	Translates SHAP explanations into linguistic summaries
2023 (Angelotti and Díaz-Rodríguez 2023)	Synthesize arena game (RL task)	Reinforcement learning with Myerson values
Joint prediction and explanation		
2018 (Hind et al. 2019)	Tic-tac-toe game, Loan repayment (classification)	Explanation with output as a single label
2018 (Park et al. 2018)	Visual question answer, Action explanation from the image	Answer a question posed with an image with textual and visual information
2016 (Hendricks 2016)	Specimen prediction (classification)	RL explains the reasons for each class
2017 (Zhang et al. 2018)	Animal images classification	With specific loss function for explanations
2018 (Chen 2019)	Specimen, type of car prediction (classifications)	Dissects the image by finding prototypical parts
Explainability through regularization		
2018 (Wu et al. 2018)	Medical diagnosis, Stop phonemes categorization	Penalize model to weak simulations
2019 (Ghaeini et al. 1902)	Event Extraction, Cloze-Style Question Answering	Model to avoid meaningless statistical biases in the data
Other methodologies		
2017 (Angelino et al. 2018)	Bail outcomes, Founding of weapon prediction	Model with optimal rule lists
Post-hoc models		
2016 (Ribeiro et al. 2016)	Product review prediction (classification)	Model added perturbed samples in the dataset
2017 (Lundberg and Lee 2017)	Written digital prediction, Medical diagnosis (classifications)	Used Shapley values to explain the contributions
2023 (Dandolo et al. 2023)	Classification of types of glass	Used quantiles of the empirical distribution of each feature

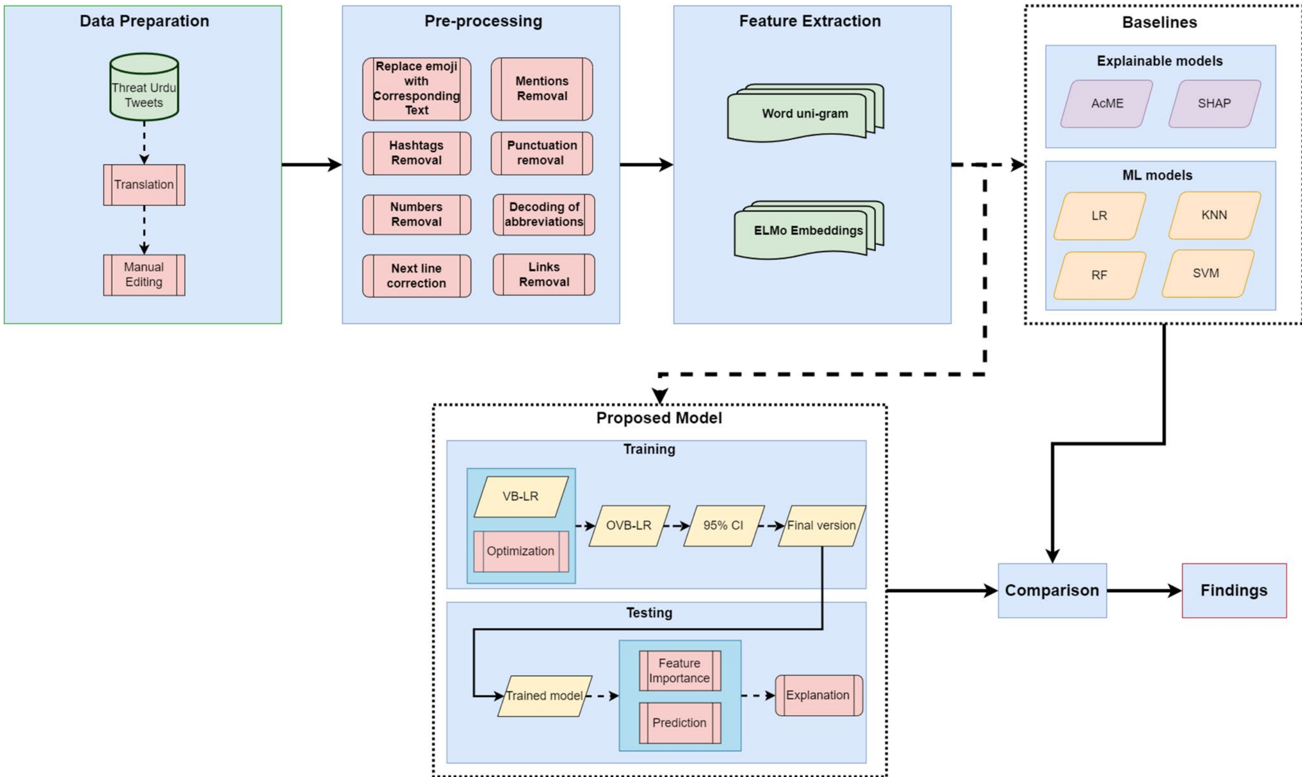


Fig. 1 Architecture of the proposed framework

pre-processing, feature extraction, proposed model training and testing, baseline experiments, result comparisons, and conclusion.

3.1 Problem statement and formulation

The problem statement can be defined as; prior studies for threatening comment detection on social media platforms did not address the issue of interpretability of prediction inference. Furthermore, only one dataset is available but have issues of inappropriate annotation.

This study aims to detect threatening speech posted on the Twitter platform. The corpus consists of two categories (Threat and non-threat). Mathematically, the problem of detecting threatening comments can be formulated as; the corpus consists of n tweets and there are a pairs of components (x_1, y_1), (x_2, y_2), (x_3, y_3), ………., (x_n, y_n). Where x_i is an ith tweet and y_i is the corresponding label (1 for threat, 0 for non-threat). The objective is to design a model f: $X \rightarrow Y$, that can predict the class label y_i €{0, 1} for each x_i with explainability. The $y_i = 1$ represents the threat class and $y_i = 0$ represents the non-threat class.

3.2 Data preparation

To test the effectiveness of the proposed OVB-LR model for threat comment identification on the Twitter platform, we designed a new dataset. Currently, there is only one dataset available for identifying threats in English (Hammer et al. 2019), but after detailed analysis, this dataset has the following issues: (1) Inappropriate annotation of YouTube comments, e.g. Offensive and abusive comments are also labeled as threats, (2) A lot majority of comments are very short in length (4–5 words), thus do not imply proper context of threats, and (3) The focus of this dataset is mainly on violence incitation not on threats (violence incitation is a special case of threats). Due to these issues, we have decided to adopt other options and explored datasets that have been created for other languages but for the same task.

The dataset proposed by Malik et al. (2023a), consists of 2400 instances, in which 1200 are threatening and 1200 are non-threatening tweets in Nastaliq Urdu. The authors collected this dataset using Twitter API and a special lexicon of seed words is used to find relevant tweets. The data was collected between August 2020 and August 2022, during the unstable political situation in Pakistan. We have chosen this dataset because its annotations are correct and achieved Fleiss' kappa inter-annotator agreement of 80%. In addition, it is a balanced dataset. This dataset is in the Urdu language, so the first step in preparing our dataset involves translating tweets from Urdu to English using the Google Translate API for automatic translation. The inclusion of this step is possible because the original dataset was created using only tweets that did not contain words from the other languages. The Google Translate API was chosen because it is one of many with an open API and supports Urdu translation.

To ensure accuracy, manual efforts were applied and a translation check was conducted by a native Urdu speaker proficient in English. Any inaccurate translations were corrected to match the original as closely as possible. This step was necessary because there were tweets in which automatic translation tended to eliminate toxicity and produced a translation that may not accurately reflect the original text (Dale et al. 2109). This process resulted in the final form of the dataset containing an equal number of instances for both classes.

3.3 Pre-processing phase

Pre-processing is a crucial step in automatic text classification, as it filters out unnecessary information and helps to extract relevant information from social media content. To prepare the data for classification tasks, especially when we have to explain the classification results, it is important to minimize inconsistencies in the data before classification. To achieve that, it was decided to perform these steps:

1. Removal of all the hashtags, HTML tags, mentions, punctuations, and URLs.
2. Removal of numbers. The step 1 and 2 are needed to lower the number of unique and meaningless words.
3. Decoding of abbreviations (thnx, thx, btw, pls, plz, etc.).
4. Replacing emoji/emoticons with corresponding text they represent. This step is needed because Emojis are important in defining the sentiment of the text.

Three additional steps are applied for the n-gram features:

1. Transform text to lowercase.
2. Lemmatization.
3. English stop-words removal.

3.4 Feature extraction

After pre-processing, two types of features are extracted to investigate their impact on the classification of threatening comments in tweets. The features are word n-gram and ELMo embeddings.

3.4.1 Word n-gram features

An n-gram model can be used to identify a unique sequence of n-words. Despite their simplicity, these models have shown significant performance in classification tasks (Malik et al. 2023b; Malik et al. 2024a; Younas et al. 2023). In this article, the word uni-gram is used to generate features to identify threat comments.

Initially, word uni-gram generated 3284 features. After that, the top-80 uni-grams are selected out of 3284 features produced by the model. This was done to reduce feature space and to concentrate on some important features. The process also helped us to simplify the explanations of feature importance. The top-80 features are chosen using the RF model.

3.4.2 ELMo embeddings

ELMo is a word embedding method that represents a sequence of words as a corresponding sequence of vectors. Unlike fixed word embeddings, ELMo considers the entire sentence before assigning each word to its embedding (Malik et al. 2023). For the experimental setup, an ELMo model trained on a News corpus is used.

Like word uni-gram, 1024 features are generated by the ELMo model. Then the study selected the top-100 important features out of 1024 using the RF model. However, we tried several values (ranging from 50 to 100) to select the top influential features and the 100 threshold resulted in the most effective list. The objective was to reduce the computational efforts (processing time) and to achieve optimal performance.

3.5 Classification and explainability

Several algorithms attempted to present the interpretability of results after the model has been trained, such models include SHAP, AcME, LIME, etc. However, these algorithms have some drawbacks:

1. They have been found to be computationally slow due to the cost of constructing and analyzing additional parameters.
2. These models provide explanations based on assumptions, which do not guarantee the accuracy and truthfulness of the results.

In contrast, other models aim to be inherently explanatory. This type of models is preferable. There are multiple ways to design such models: 1) developing a straightforward model that is easily understandable, 2) combining multiple models where one clarifies the results of the others, and 3) providing the model with data that contains an explanation of the results, etc. An example of such models (inherently explainable) is our proposed model (OVB-LR), which offers a solution for improving the performance and interpretability of prediction inference.

3.5.1 Variational Bayesian logistic regression

As described earlier, this study develops a robust framework that aggregates feature importance, intrinsic interpretability mechanism, and impact of salient features to improve performance and interpretability. The proposed framework explores the relationship between important features and intrinsic interpretability. It combines the strength of Bayesian logistic regression, and variational inference with optimality. A brief description of each component of the proposed framework is provided below:

Logistic Regression with Bayesian paradigm: LR is one of the conventional ML models used for the task of classification. It is a probabilistic model and an extension of a linear regression model in which the probability of success can be estimated by taking a sigmoid of linear transformation of features.

Bayesian modeling is to learn posterior distribution given the prior distribution of the data and observed parameters. Therefore, to build a model in a Bayesian fashion, we need to formulate the generative process that generates the observed data.

Variational inference: In the current era, obtaining large volumes of labeled and high-quality data is still expensive. Working with small annotated datasets and high dimensions leads to serious over-fitting problems. Variational inference can deal with this issue effectively. It can handle the issue of accurately calculating the posterior probability of latent parameters in the presence of a small dataset size by using the simple distribution. Thus, variational inference mitigates the factor of overfitting by employing a prior distribution to approximate an optimal logistic regression model.

OVB-LR: So this model uses a variational inference mechanism to estimate the regression coefficients that are used to highlight the significance of each feature, as well as help in the classification process. This is achieved by using a mechanism to approximate the posterior probability distribution. These features also make the OVB-LR model useful in the presence of small datasets, because it helps accurately determine the posterior probability of latent variables of the model.

By using variational inference to learn model parameters, it is possible to calculate a 95% confidence interval for the regression coefficients using the mean, covariance of the approximate posterior probability distribution, and z-value of the 95% confidence interval in the standard normal distribution. These values are helpful to identify the salient features contributing to the predictive output. The first feature in the descending sequence of mean weights, identified with different numbers of upper and lower bounds within the 95% CI, serves as a boundary. Features with an equal number of upper and lower bounds before the boundary are classified as salient, meaning they are the most important features. If the coefficient estimate falls outside the 95% CI, positive values indicate a feature with a positive influence, while negative values indicate a feature with a negative influence.

The OVB-LR model can be tested for the classification task using small sample datasets (Liu et al. 2024). The model provides interpretability to its predictive inference by approximating the posterior probability. This estimation facilitates the selection of significant features by assessing their weight and impact on the model's results. The stability of the OVB-LR model is also tested and analysis concluded that it is a better stable model compared to other models. The OVB-LR model determines features' importance, which is comparable with other post-hoc explainable methods' output. The methods are SHAP, AcME, etc. The pseudo-code of the learning process for the OVB-LR model is presented in Algorithm 1.

The description of the Algorithm 1 is provided below.

Input parameters: training dataset (x_{train}, t), maximum number of iterations (iter_max), list of features from the dataset (feature_names), hyper-parameters of the model, which are defined by the user (a_0, b_0).
Step 1: Initialization of parameters with values provided by the user for variables $(a_N, b_N, param)$.
Step 2: Execute all steps inside the loop with iterations equal to iter_max.
Step 3: Update a_N, b_N parameters using weights, which are obtained from the previous iteration, to calculate a mean of the Gamma distribution.
Step 4: Update the covariance matrix of the approximate Gaussian distribution.
Step 5: Update the mean vector of the approximate Gaussian distribution.
Step 6: Update variational parameters ξ. If the difference between the current ξ and the previous ξ (param) is minimal, then we stop the algorithm, otherwise, we continue it.
Output parameters: Trained weights of the model.

Parameter optimization: To obtain the optimal values of hyper-parameters α, regression coefficients, and latent parameters $\boldsymbol{w}$, the steps of Algorithm 1 need to be executed sequentially by updating the $a_N, b_N, \mu_N, \Sigma_N$ and ξ iteratively. The Σ_N and ξ are updated until the condition on step 7 (Algorithm 1) fulfills or up to the iter_max (maximum iteration of loop at step 2). Here, μ_i, Σ_i represents the mean and variance of the regression coefficients of the ith characteristic parameter. We use the statistical measure mean to describe the characteristic significance and its impact and standard deviation to pinpoint its stability.

Thus OVB-LR framework provides a benchmark solution for high performance and intrinsic interpretability for threatening comment detection by utilizing prior knowledge through a Bayesian approach and effectively addresses the issue of data sparsity. The proposed framework is tested on a newly designed threatening comment corpus in Sect. 4 and the conclusion is drawn.

3.6 Experimental setup

This section describes the baseline ML models and explainable models that are chosen to compare the performance of the proposed framework in classification and explainability tasks. The reason why we chose these ML models is because they have demonstrated significant performance for Urdu threat comments identification (Malik 2023) and other similar tasks (Nawaz and Malik 2022). For comparing explainability, SHAP (Lundberg and Lee 2017), and AcME (Dandolo et al. 2023) are chosen. The SHAP algorithm can provide explanations for the outcome of any ML or DL model on an individual example (local level) or the overall effect of a feature on the outcome (global level). It is considered a state-of-the-art independent interpretability method. The AcME is a new algorithm that explains classification and regression results at local and global levels. It has been shown to provide explanations of comparable quality to SHAP, while significantly reducing computational time (Dandolo et al. 2023). Python language is used for development purposes.

Algorithm 1 Pseudo-code of the OVB-LR—Learning process.

Require: $x_{\text{train}} \in \mathbb{R}^{N\times D}$, $t \in \mathbb{R}^N$, $iter\_max$: int, Number of iterations, $feature\_names$: list, represents a list of features, a_0 and b_0 are hyperparameters, with a default value of 1.0.
Ensure: Feature weighting sequence.

Initialization: ξ is randomly sampled,
  $\boldsymbol{param} \leftarrow \xi$, $a_N = a_0$, $b_N = b_0$.
for _ **to** $iter\_max$ **do**
  update:
    $a_N = a_0 + \frac{D}{2}$
    $b_N = b_0 + \frac{1}{2}\mathbb{E}[\boldsymbol{w}^{\mathrm{T}}\boldsymbol{w}]$
    $\mathbb{E}[\alpha] = \frac{a_N}{b_N}$
  update $\boldsymbol{\Sigma}_N$,
    $\lambda(\xi_n) = \frac{1}{2\xi_n}[\sigma(\xi_n) - \frac{1}{2}]$,
    $\boldsymbol{\Sigma}_N^{-1} = \mathbb{E}[\alpha]\boldsymbol{I} + 2\sum_{n=1}^{N} \lambda(\xi_n)\phi_n\phi_n^{\mathrm{T}}$,
  update $\boldsymbol{\mu}_N$,
    $\boldsymbol{\mu}_N = \boldsymbol{\Sigma}_N \sum_{n=1}^{N} \left(t_n - \frac{1}{2}\right)\phi_n$
  update ξ,
    $\xi \leftarrow \xi_n^{\text{new}} = \sqrt{\phi_n^{\mathrm{T}}(\boldsymbol{\Sigma}_N + \boldsymbol{\mu}_N\boldsymbol{\mu}_N^{\mathrm{T}})\phi_n}$
  if $\|\xi - \boldsymbol{param}\|_2 < 10^{-5}$ **then**
    output and **break.**
  else
    $\boldsymbol{param} \leftarrow \xi$.
  end if
end for

3.6.1 Baseline classifiers

This section outlines the machine learning models used as a baseline for the classification task. The models are LR, KNN, SVM, and RF.

3.6.1.1 Logistic regression Logistic regression is a supervised machine learning model. In its basic form, this model uses logistic functions to predict the probability of a binary outcome and has demonstrated benchmark performance in text-mining tasks (Abbas and Malik 2023). For the implementation of LR, the sklearn library was used with default parameters.

3.6.1.2 K-nearest neighbors K-nearest neighbors are also a supervised learning method. The model categorizes an object into the class that appears most frequently among its k-nearest neighbors. One of the algorithm's key features is that it does not make any underlying assumptions about the distribution of data. It has proved his effectiveness in related tasks (Mehboob and Malik 2021). For the implementation of KNN, the sklearn library was used with default parameters.

3.6.1.3 Support vector machine The support vector machine is one of the conventional supervised learning techniques. The main logic is to construct the hyperplanes that optimally separate the sample objects. The algorithm operates under the assumption that the larger the distance between the separating hyperplanes and the objects of the separated classes, the smaller the average error of the classifier will be. This model demonstrated state-of-the-art performance in NLP tasks (Malik and Nawaz 2024; Malik et al. 2024b). The sklearn library was used with default parameters for the coding of this algorithm.

3.6.1.4 Random forest The random forest is an ensemble model that is based on the bagging approach. The ensemble mechanism consists of multiple decision trees. Each tree classifies an object into one of the classes, and the final class is—determined by the majority of the obtained classes. It showed robust performance in NLP and text mining tasks (Malik et al. 2023c; Ali and Malik 2023).

3.6.2 Baseline explainable models

This section describes the baseline explainable models that were used for the comparison with the proposed OVB-LR model.

3.6.2.1 Shapley additive explanations Shapley Additive Explanations is an interpretable AI method that uses a game-theoretic approach to explain the output of any machine learning model. In this method, each feature is treated as a 'player' in a prediction game, and the method measures each player's contribution to the final outcome by using Shapley values. The Shapley value is a concept used in game theory that involves fairly distributing both gains and costs to players. We used SHAP as a baseline to compare the interpretability of the proposed model. For implementation in Python, the shap library was used with the default parameters of Explainer.

3.6.2.2 Accelerated model-agnostic explanations Accelerated Model-agnostic Explanations is an interpretability approach that provides feature importance scores, both globally and locally. AcME estimates importance which is derived from perturbations of the data using quantiles of the empirical distribution of each feature. This method is also chosen as a baseline for comparison. For the implementation of AcME, the statwolf AcME library was used.

3.6.3 Evaluation metrics

The results are analyzed using the following metrics:

- Accuracy $= \frac{TP+TN}{TP+TN+FP+FN}$
- F1-score $= \frac{TP}{TP+\frac{1}{2}(FP+FN)}$
- F1-score macro-averaged $= \frac{\sum_{i=1}^{N} (\text{F1-score})_i}{N}$
- F1-score weighted-averaged $= \frac{\sum_{i=1}^{N} (\text{F1-score})_i * \text{support}_i}{\sum_{i=1}^{N} \text{support}_i}$

True Positive (TP): Samples that are positive and predicted correctly as positive;

False Positive (FP): Samples that are negative but predicted incorrectly as positive;

False Negative (FN): Samples that are positive but predicted incorrectly as negative;

True Negative (TN): Samples that are negative and predicted correctly as negative.

N: Number of classes.

Support: Number of instances of one class.

We selected accuracy metric because the dataset contains an equal number of examples for both classes. The f1-score is suitable in general as it considers both positive and false negatives, providing an overall view of misclassification. The macro-averaged f1-score is appropriate in this case as it provides an aggregate f1-score for both classes.

4 Results and analysis

This section presents experiments to evaluate the effectiveness of two types of features with the OVB-LR model for threat comment identification. The OVB-LR classifier is compared with four traditional ML classifiers to define the best model. Additionally, experiments are performed to evaluate the interpretations of OVB-LR and their comparison with SHAP and AcME XAI models.

4.1 Comparison of predictive performance

This section describes and compares the results of experiments conducted using word uni-gram and ELMo embeddings to classify dataset instances into threat or non-threat classes. For this purpose, the OVB-LR model and four ML models, including LR, KNN, SVM, and RF are used for the experiments. For all the experiments, only the important features are chosen using the RF model. This was done because word uni-gram and ELMo models produce more than 1000 features, which makes interpretability quite difficult.

Table 4 shows the results of the experiments conducted using uni-gram features. We used top-80 features in the experiment. The results show that the OVB-LR model outperforms other models in terms of accuracy, f1-score for non-threat class, macro-average f1-score, and weighted f1-score. However, for the f1-score of the threat class, the OVB-LR model has comparable performance to SVM and RF. We observed significant improvement in macro and weighted f1-score demonstrated by the OVB-LR model compared to baselines. The largest improvement is observed for non-threat classification. Since the dataset on which the experiments are performed is balanced, the accuracy metric is suitable for comparing the performance of the models. In summary, the OVB-LR model with unigram features outperformed the baselines (four ML models).

The SVM model achieved slightly better performance than the OVB-LR model for only threat class detection. This can be explained by the fact that it can handle non-linear separable data using kernel functions and is able to find the decision boundary that maximizes the margin between different classes. This factor is believed to improve the model's generalization performance. Because of this, SVM is more flexible in terms of class predictions compared to OVB-LR model, which is based on logistic regression. However, interpretation of the results is more complicated due to the use of multiple spaces and the OVB-LR model despite lagging behind (in terms of f1-score for threat class), supports interpretable prediction.

To investigate the impact of hyperparameters (a0 and b0) on the accuracy of OVB-LR model and to determine

Table 4 Comparison of classifiers using word unigrams (top-80)

Classifiers	Accuracy	F1-score			
		Threat	Not-threat	Macro-AVG	Weighted-AVG
KNN	80.00	79.13	80.80	79.97	79.97
SVM	80.42	80.50	80.33	80.42	80.42
RF	80.00	79.83	80.17	80.00	80.00
LR	79.58	78.97	80.16	79.57	79.57
OVB-LR	80.83	79.46	82.03	80.75	80.75

the maximum achievable accuracy, tunning of hyperparameters (a0 and b0) is performed. The results of this experiment are presented in Fig. 2. It is evident from results that the highest accuracy (80.83) is achieved when the value of a0 increases (from 10 to 100) and b0 is in the range of 10 and 30. Conversely, the lowest accuracy (79.60) is obtained when b0 approaches to 1 and a0 is above 85. This analysis guided us to choose the optimum values for a0 and b0 hyperparameters.

Next, the confusion matrix is shown in Fig. 3 which is obtained using the top-80 word uni-gram features. Among the 120 threatening examples, the OVB-LR model accurately classified 89 samples (TP) and misclassified 31 (FN). Among the 120 non-threat examples, the OVB-LR model accurately classified 105 samples (TN) and misclassified 15 (FP).

The next experiment is carried out to explore the impact of features generated by the ELMo model on threat content identification and results are added in Table 5. The top-100 features are chosen for the classification and explainability tasks due to the large number of features (i.e. 1024). The results indicate that the OVB-LR model has the best performance compared to other models in terms of accuracy, f1-score for threat and non-threat classes, macro and weighted-average f1-scores. In addition, ELMo features achieved better metric values compared to word uni-gram for classification task. We observed substantial improvement along all metric values with ELMo + OVB-LR configuration compared to baseline ML models. The second-best performance is observed with ELMO + RF configuration.

To investigate the impact of hyper-parameters (a0 and b0) on the accuracy of the OVB-LR model in the presence of ELMo features and to determine the maximum achievable accuracy, tuning of hyper-parameters (a0 and b0) was performed. The results of this experiment are presented in the Fig. 4, and it is clearly visible that the highest accuracy (81.2%) was achieved with small a0 and significantly large b0 values. Conversely, the lowest accuracy values (67.5%) are observed with quite large a0 values and with b0 close to 1.

To visualize the components of confusion matrix generated by the OVB-LR + ELMo model, the confusion matrix is shown in Fig. 5. Among the 120 threat test samples, the OVB-LR model accurately classified 95, while 25 are classified incorrectly. Likewise, the OVB-LR model accurately classified 100 out of 120 non-threat test samples, while 20 are classified incorrectly. By comparing two feature models (word uni-gram and ELMo) and five ML models for the threat comments identification task, we achieved 81.25% accuracy and 81.24% macro and weighted f1-scores with ELMo + OVB-LR model as the best performance. Thus it is established that the OVB-LR model is better than the four ML models for threat comments identification task.

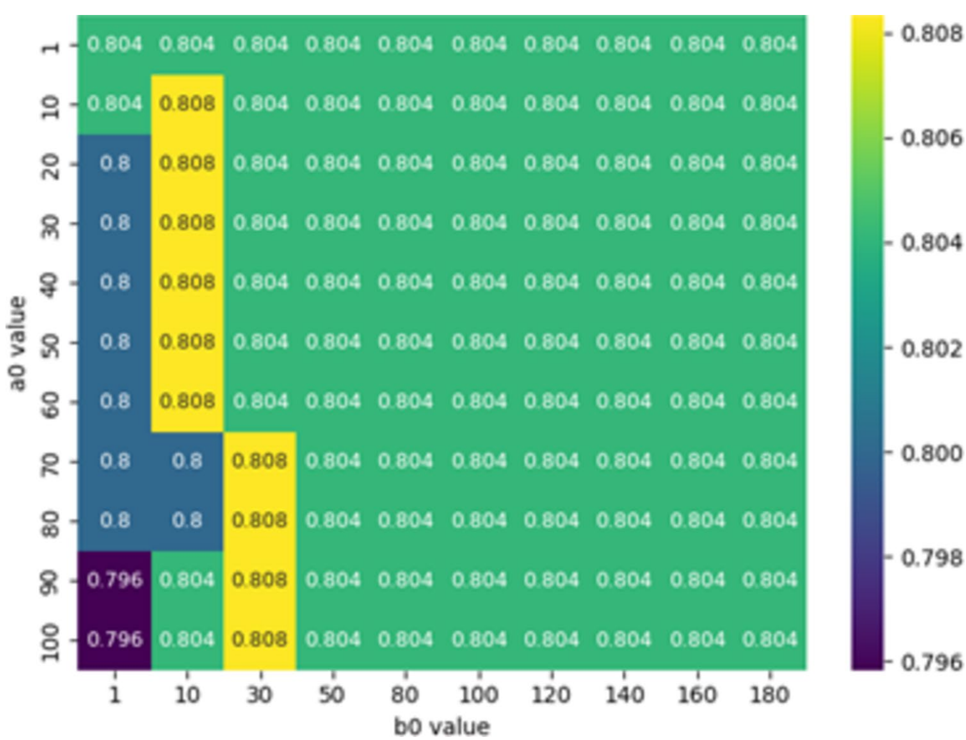


Fig. 2 Hyperparameters (a0 and b0) tuning results for OVB-LR model (Unigram features). Values in boxes represent the accuracy achieved by specific parameters

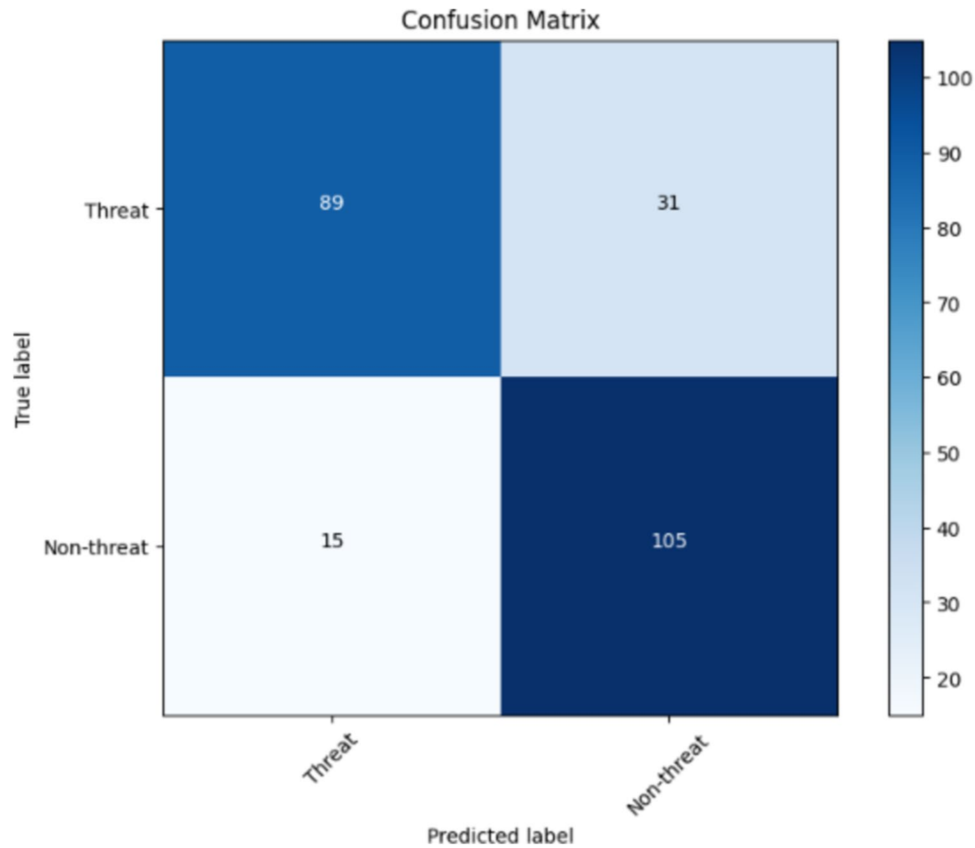


Fig. 3 Confusion matrix for OVB-LR model (Unigram features). The horizontal axis represents the predicted labels and the vertical axis represents the true labels

Table 5 Comparison of classifiers using ELMo features (top-100)

Classifiers	Accuracy	F1-score			
		Threat	Not-threat	Macro-AVG	Weighted-AVG
KNN	77.08	79.40	74.18	76.79	76.79
SVM	80.42	80.33	80.50	80.42	80.42
RF	80.83	80.83	80.83	80.83	80.83
LR	76.67	77.24	76.07	76.65	76.65
OVB-LR	81.25	80.85	81.63	81.24	81.24

4.2 Interpretability validation

This section presents the results of experiments conducted for explainability using proposed OVB-LR and compares with state-of-the-art AcME, and SHAP XAI models. The word uni-gram and ELMo features are utilized for the experimental setup to classify dataset instances into threat or non-threat classes.

4.2.1 Interpretability validation on uni-gram features

The results of importance calculated by the OVB-LR model on top-80 word uni-gram features are presented in Table 6. It is visible that the three most important features are: *listen*, *stop*, and *eyes*. All of the features in the top-20 list are salient and have a positive impact on the prediction of threat class. The salient features are those that play a significant role in the prediction of positive class.

The most important feature for the prediction of threat class according to the OVB-LR model is *listen*. This can be explained by the fact that the 'listen' word can be used in the following context:

1. The beginnings of threatening. If the individual or group to whom the message is directed does not pay attention to the words, then the culprit can get worse. For example, "Listen, you will die on me or I will kill you by strangling you".
2. The individual or group to whom the message is addressed, listens to what the author considers to be an incorrect source of information. For example, "Don't listen to nonsense and shut your stupid mouth"
3. Pointing someone to listen. For example, "Let all the Jewish Christians open their ears and listen."

The description of attributes of Table 6 is presented next: The second column shows the names of the features and the third contains the mean and standard deviation of the feature regression coefficients. The fourth and fifth columns are the corresponding upper and lower bounds of the 95% CI of the

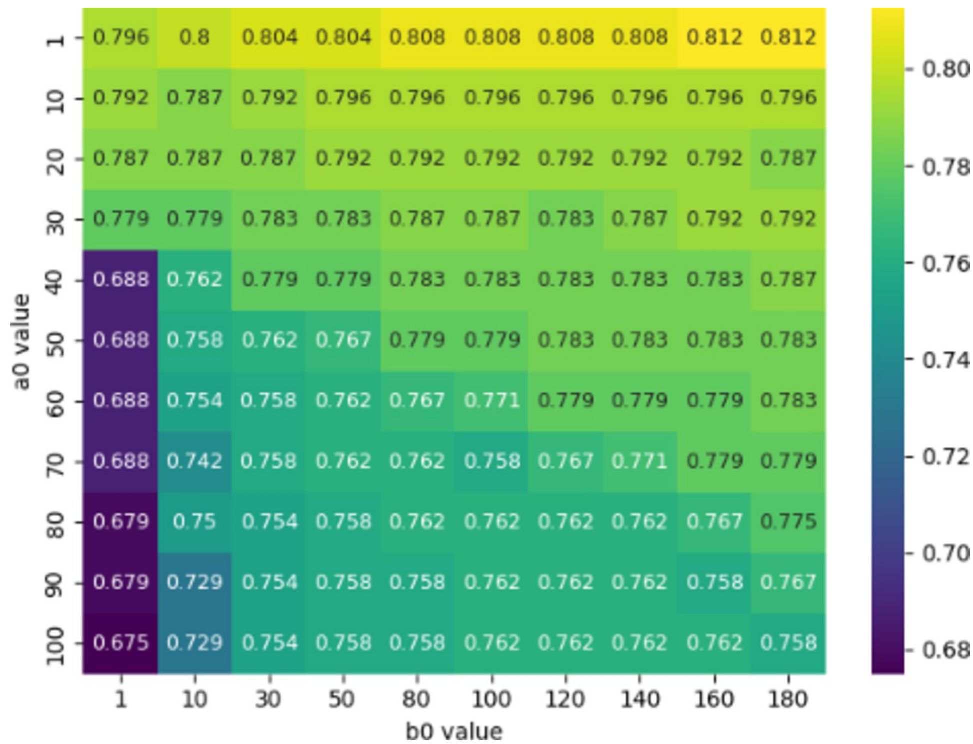


Fig. 4 Hyperparameters (a0 and b0) tuning results for OVB-LR model with ELMo features in accuracy

regression coefficients. The sixth column specifies whether the feature is salient or not, indicating that it influences the prediction either toward threat or non-threat classes (upper and lower bounds have the same sign). Thus OVB-LR model highlighted that these twenty features are salient and play a significant role in prediction.

Considering baselines, the ranking of word uni-gram features based on mean absolute SHAP values is presented in Fig. 6. The graph provides evidence of the overall influence of each feature on model prediction for threat and non-threat class instances. The graph does not show dramatic changes in the importance of the features and does not show any feature that significantly affects the model's outcome. This suggests that no specific attributes are likely to play an important role in model prediction according to SHAP. Moreover, it is clear that the cumulative sum of SHAP values across multiple attributes, rather than any individual attribute, is crucial in determining the model results.

Next, the results of ranking proposed by SHAP values using a summary plot are presented in Fig. 7. The ranking indicates the correlation between object values and their impact on model output. The vertical axis contains feature names and ranks the features from top to bottom based on importance. For the horizontal axis, positive SHAP values (red color) indicate a positive effect on model prediction (classification to threat class), whereas a negative one (blue color) indicates a negative effect on model prediction (classification to non-threat class). Color is a representation of positive or negative impact on prediction: pink is the highest value, and blue is the lowest value. Based on the presented results, the contribution of each feature is shown in terms of positive and negative values (red and blue color) and its impact on the model prediction (threat class). According to the summary plot, the three most significant features are: "*kill*, *khan*, and *stop*".

Additionally, according to the SHAP model, '*kill'* is the most important feature for predicting the threat class. This can be explained by the fact that the use of violent language is present, as it includes a threat of killing (e.g. "Indians, we will kill you even if we have to give our lives for our country"). On the other end, the feature '*imran'* has a negative effect on prediction (non-threat class).

Next, the second baseline model for comparison is AcME and the results are shown in Fig. 8. The bar graph indicates that the most important feature in determining the threat class is '*khan'*, which is part of the name of the 'Pakistani politician Imran Ahmed Khan Niazi', whose name appears in the dataset as a victim. The following most important features are '*India* and *lesson*'. Starting from the 4th feature ('Pakistan'), there is a sharp drop in importance for determining the threat class of an object.

In conclusion, the proposed OVB-LR, SHAP, and AcME models share a similar set of important features. However, each model indicates three most important features that significantly influence the classification of the threat instances:

1. The OVB-LR model ranked the "*kill*" feature as the 4th most influential in predicting threat class, whereas the

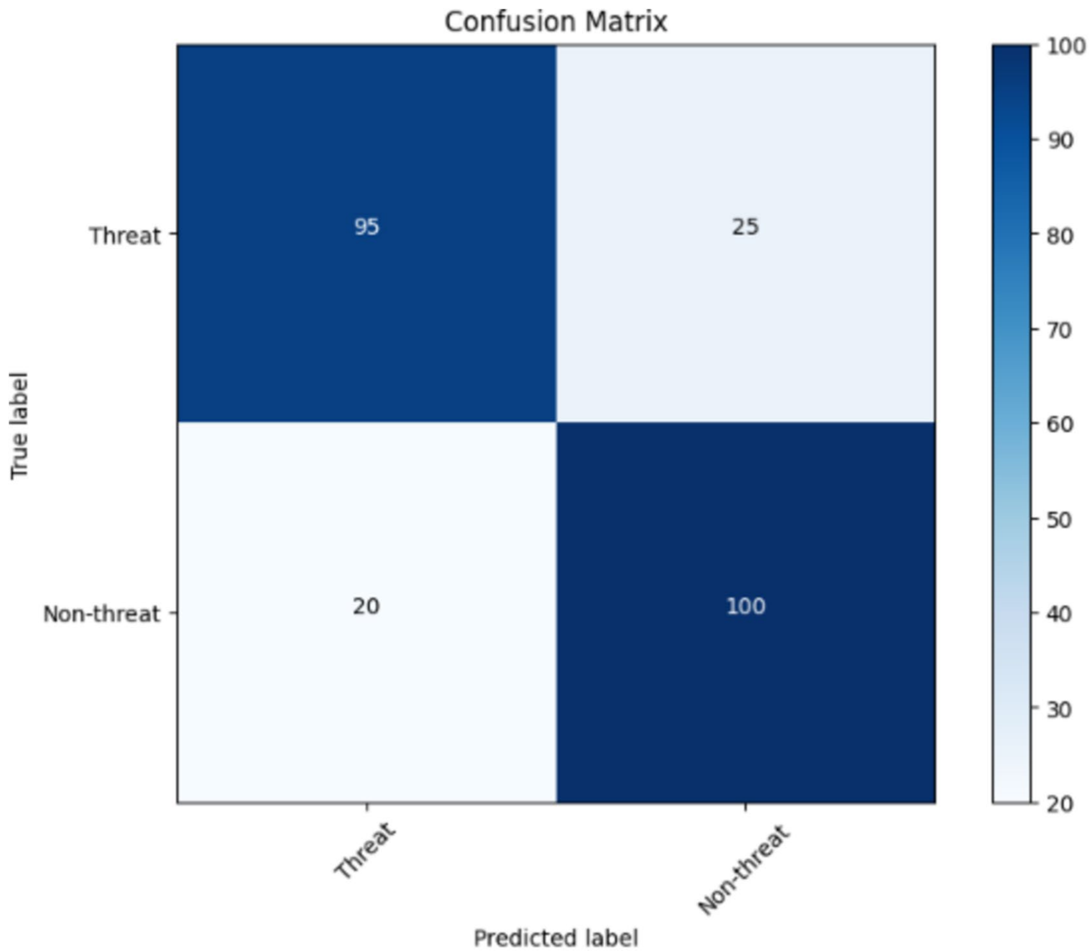


Fig. 5 Confusion matrix for OVB-LR Model (ELMo features)

SHAP model identified this feature as the most important, and AcME ranked this feature as the 10th most important.

2. The AcME model identified the "*khan*" feature as the most important. The OVB-LR model ranks this feature at 9th position, whereas the SHAP model places the "*khan*" feature in 2nd place and states that it has a positive impact on threat class prediction.
3. The OVB-LR model highlights the "*listen*" feature as the most significant attribute, whereas the AcME model places the *listen* feature in the 8th position. The SHAP model ranks this feature in the 6th position and states that it has a strong positive effect on threat class prediction.

4.2.2 Interpretability validation on ELMo features

The objective of the next experiment is to evaluate the explainability provided by the OVB-LR model using ELMo embeddings and its comparison with two state-of-the-art SHAP and AcME post-hoc models. The ranking of the top 20 ELMo features proposed by the OVB-LR model is presented in Table 7. The ELMo model generates 1024 features and all features are not equally important, therefore we selected the top-100 features in the classification task (Table 5). We have investigated the relationships between top-20 ELMo features (ranked by the OVB-LR model) and corresponding words from the corpus. For this purpose, we have provided words of our dataset to the ELMo model step by step to get their sole vector representations. Then we selected only those words with the highest scores corresponding to selected ELMo features. The corresponding words related to each ELMo feature are added in Table 7.

The three most important features are, "*feature 361 (khan), feature 633 (Pakistan), and feature 532(army)*". All the features up to the 8th position and 'feature 952 (listen)' are salient (important) for classification. Furthermore, the top 20 features except for "*feature 60 (tear), feature 51 (burn), and feature 849 (kill)*" have a negative impact on the model prediction, that's why we got a higher f1-score for non-threat classification compared to threat classification using ELMo features (Table 5).

Table 6 Feature importance [using top-80 unigrams] proposed by OVB-LR model

S #	Name	Weight	Lower bound	Upper bound	Is_salient_feature
1	Listen	0.64 ± 0.1267	0.3917	0.8883	True
2	Stop	0.4966 ± 0.0858	0.3284	0.6648	True
3	Eyes	0.4457 ± 0.0892	0.2709	0.6205	True
4	Kill	0.4335 ± 0.0861	0.2647	0.6023	True
5	Tear	0.4278 ± 0.0943	0.2430	0.6126	True
6	Shut	0.4083 ± 0.0893	0.2333	0.5833	True
7	Brick	0.3963 ± 0.1068	0.1870	0.6056	True
8	Disgraced	0.3755 ± 0.0955	0.1883	0.5627	True
9	Khan	0.3732 ± 0.1001	0.1770	0.5694	True
10	Shoot	0.3715 ± 0.0884	0.1982	0.5448	True
11	Anyone	0.3712 ± 0.0925	0.1899	0.5525	True
12	Threatened	0.3683 ± 0.08	0.2115	0.5251	True
13	Lesson	0.3443 ± 0.1206	0.1079	0.5807	True
14	Hang	0.3283 ± 0.1116	0.1096	0.5470	True
15	Difficult	0.3272 ± 0.0763	0.1777	0.4767	True
16	Teeth	0.3259 ± 0.0901	0.1493	0.5025	True
17	teach	0.3242 ± 0.119	0.0910	0.5574	True
18	Break	0.3148 ± 0.0987	0.1213	0.5083	True
19	Stupid	0.3014 ± 0.0966	0.1121	0.4907	True
20	Legs	0.2967 ± 0.0877	0.1248	0.4686	True

We have compared the ranking of top-20 word uni-gram and ELMo features (generated by the OVB-LR model) and the results are shown in Table 8. Considering the unigram, all features are salient, and "listen" is the most significant feature and has the strongest positive impact on the prediction. Furthermore, most of the unigrams are kind of threatening words used in the communication e.g. kill, tear, shoot, stop, etc. Another important point is that top-20 unigrams have a positive impact on the prediction, contributing positively to the prediction of threatening class. In contrast, the majority of top-20 ELMo features proposed by the OVB-LR model negatively contribute to the prediction. Only three features contributed positively to the model prediction, i.e. feature 60 (tear), feature 51 (burn), and feature 849 (kill). In addition, the positive contributing features are not salient. That's why, seventeen negative contributing ELMo features make the non-threatening class prediction higher than the threatening class prediction. Thus, top-20 ELMo features are helpful for non-threatening class prediction whereas top-20 uni-grams are helpful in predicting threatening class.

Next, the importance of features spotted by the SHAP model is presented as a summary plot in Fig. 9. It is evident that no feature has a strong correlation with model prediction. However, it is possible to divide features into two categories in general: (1) Large values of the first category positively affect model prediction, and (2) Large values of the second category negatively affect model prediction. The first category includes features: *feature 195, feature 807, feature 257, feature 401, feature 849, feature 274, and feature 149*. The second category includes *feature 361, feature 538, feature 952, feature 343, feature 633, feature 198, feature 846, feature 735, feature 1000, feature 162, feature 440, and feature 532*.

The importance of features derived by the AcME model in the form of a bar plot is presented in Fig. 10. The most important feature is '*feature 195*' in determining the threat class. The following most important features are '*feature 343* and *feature 361'*. The importance of other features gradually drops for threat class identification and *'feature 274'* is the least important in top-20. The three models show a similar set of important features. However, each model has a different top three features that significantly influence the model's classification:

1. The OVB-LR model identifies '*feature 361'* as the most important feature (salient). whereas AcME ranks it at 3rd position. The SHAP places '*feature 361'* at the 2nd position and claims that it has a stronger negative impact than a positive one on the prediction of threat class. In addition, OVB-LR also concluded that this feature has a stronger negative effect on the prediction of threat class instances.
2. The SHAP model identified '*feature 195'* as the most important, and AcME also put it in 1st position. How-

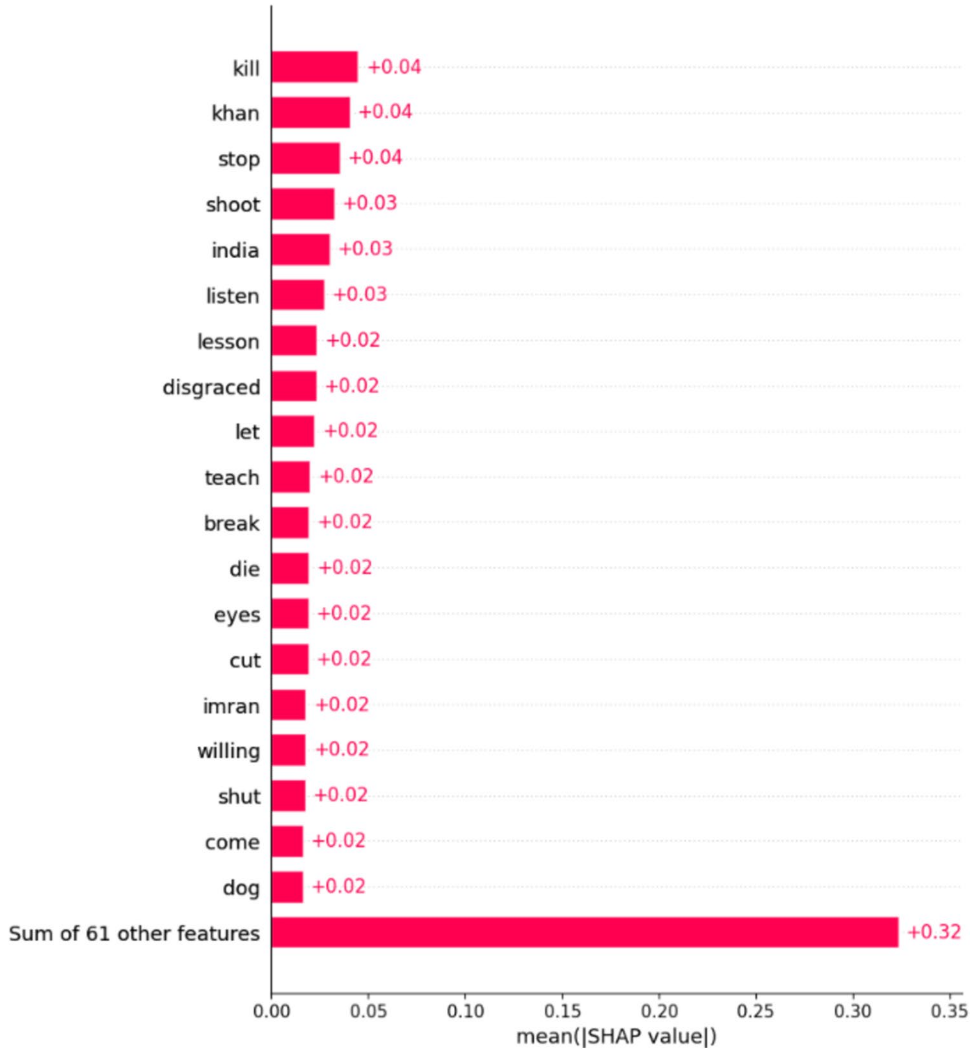


Fig. 6 Ranking of important features [unigrams] by SHAP model. Bar plot in which the horizontal axis shows the mean absolute SHAP values. The larger the value, the greater the impact on the model's prediction result

ever, the OVB-LR model doesn't put this feature in the top 20.

In the next experiment, we evaluated the explainability offered by SHAP and AcME models on one instance of threatening comments. The OVB-LR model did not support local explainability (interpretability on random instances). The SHAP force plots are presented in Fig. 11a and the estimated class for this instance is the 'threat' with a base value of 0.46 and a prediction probability of 57%. The SHAP assigns a value to each word that contributes to the prediction. The words in shades of red color positively contribute to predicting threat class whereas words in blue shades negatively impact the prediction. For example, the 'listen' word has a dark red color, indicating its strong positive contribution, likewise, 'enemies and 'Imran' have a blue color, showing a negative contribution. The importance of each word proposed by the AcME model, contributing to the prediction of class label for Tweet 1 is presented in Fig. 11b. The word 'khan' has the highest importance in prediction and AcME identified this instance as 'threat'. The word 'listen' is at the 2nd rank. Thus, both AcME and SHAP have similar important features for the given threat instance.

This completes the interpretation provided by the OVB-LR, SHAP, and AcME models for the classification of threat tweets. Thus, the proposed model (OVB-LR) provides appropriate explanations for the classification of threat comments. In addition, these explanations are comparable with SHAP and AcME explanations by considering inherently and post-hoc XAI models.

5 Discussion and limitations

In today's world, the internet has become an essential part of people's lives. Likewise, social networks have become a primary source of information for many people. They have accelerated the process of information dissemination and have enabled individuals to express their opinions on various events and incidents. These opinions can be positive,

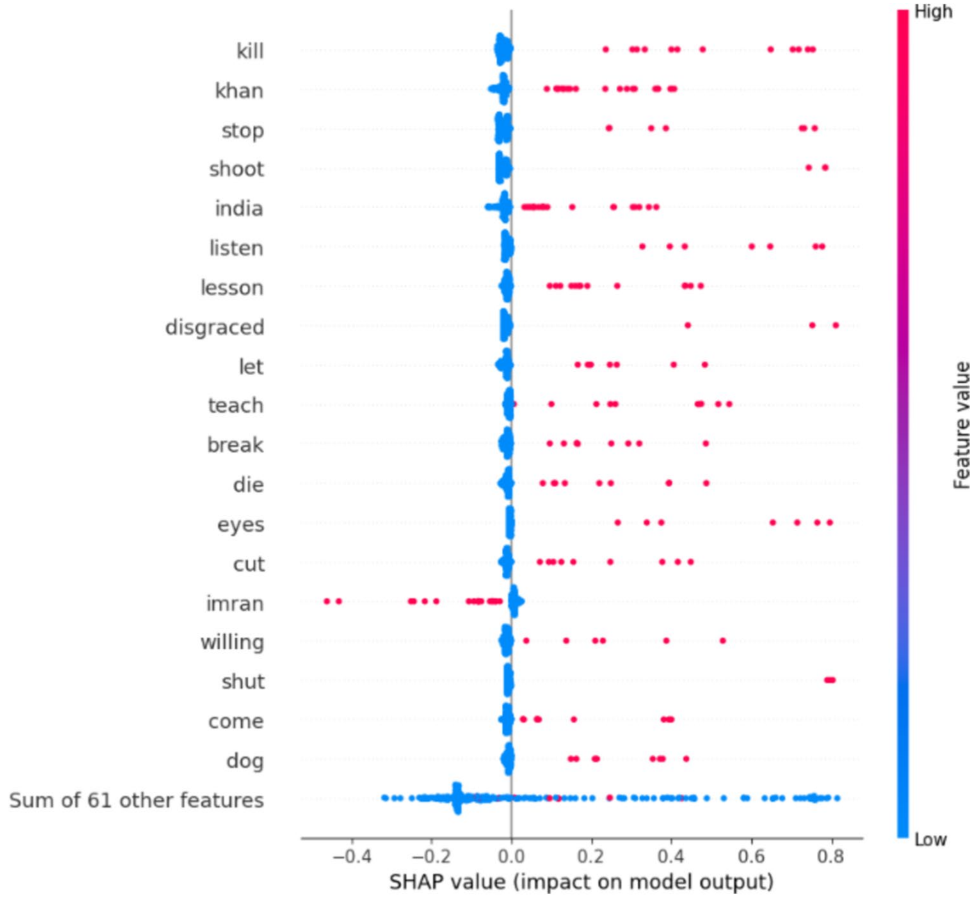


Fig. 7 Summary plot of SHAP-based feature importance [top-80 unigrams]

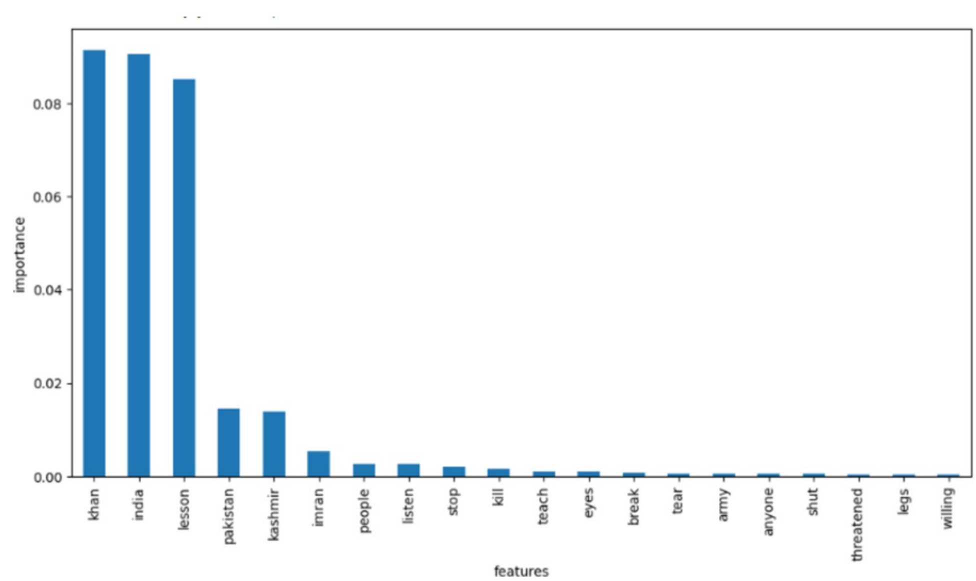


Fig. 8 Feature ranking [top-80 unigram] proposed by the AcME model. The horizontal axis shows the importance of features. The larger the value, the greater the impact on the model's prediction

neutral, or negative (Xiao et al. 2024; Xiao et al. 2022; Mao et al. 2022). However, negative comments can lead to threats from those with opposing views. Unfortunately, these platforms also became a source for online threats that target vulnerable groups based on their religion, gender, interests, etc. It is important to note that social media posts can be made publicly or privately. The initiator of threats may use their knowledge of social media, the ability to hide their identity, and the victim's limited options for defense and escape to dominate their victims. Thus, threatening can have

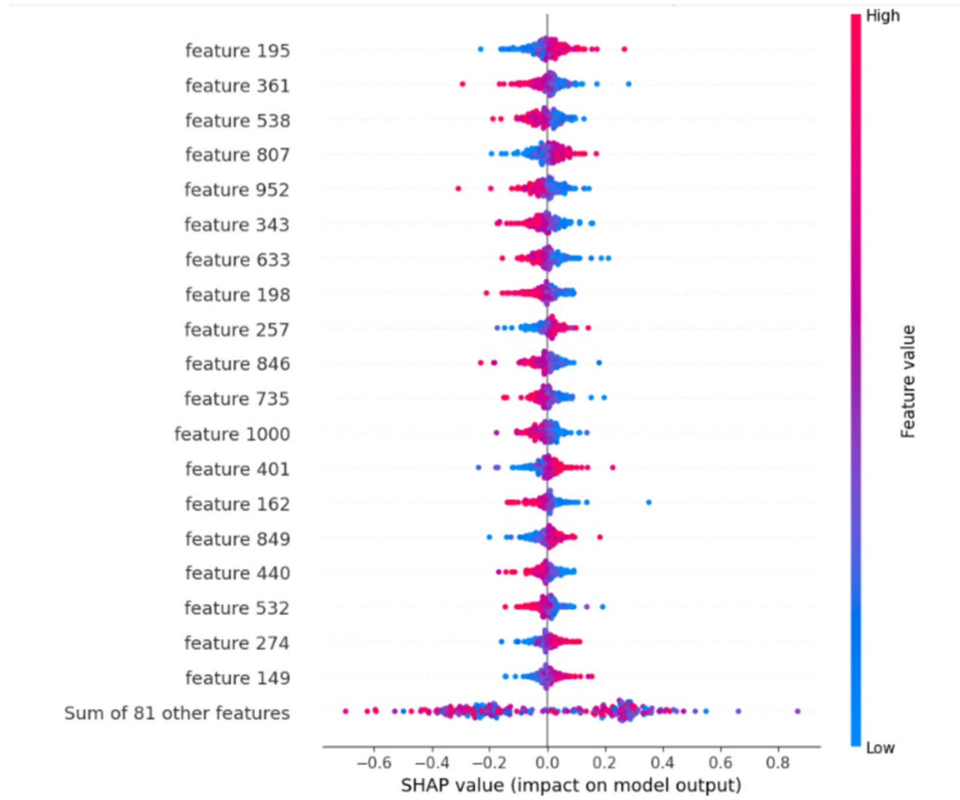
Table 7 Feature ranking [using top-100 ELMo features] proposed by OVB-LR model

S #	Name	Corresponding words	Weight	Lower bound	Upper bound	Is_salient_feature
1	Feature 361	Khan	− 10.0693 ± 3.8836	− 17.6812	− 2.4574	True
2	Feature 633	Pakistan	− 9.9991 ± 3.7884	− 17.4244	− 2.5738	True
3	Feature 532	Army	− 9.0647 ± 4.5108	− 17.9059	− 0.2235	True
4	Feature 1000	Anyone	− 8.5581 ± 3.8082	− 16.0222	− 1.0940	True
5	Feature 343	Stupid	− 8.4728 ± 3.5904	− 15.5100	− 1.4356	True
6	Feature 735	People	− 8.2892 ± 3.6098	− 15.3644	− 1.2140	True
7	Feature 846	Eyes	− 8.1079 ± 3.596	− 15.1561	− 1.0597	True
8	Feature 673	Imran	− 7.9851 ± 4.036	− 15.8957	− 0.0745	True
9	Feature 60	Tear	7.8552 ± 4.1391	− 0.2574	15.9678	False
10	Feature 606	Threaten	− 7.4519 ± 4.2112	− 15.7059	0.8021	False
11	Feature 371	Lesson	− 7.2345 ± 4.0077	− 15.0896	0.6206	False
12	Feature 941	Shut	− 6.9435 ± 4.1371	− 15.0522	1.1652	False
13	Feature 198	Stop	− 6.931 ± 3.7605	− 14.3016	0.4396	False
14	Feature 51	Burn	6.9124 ± 4.2313	− 1.3809	15.2057	False
15	Feature 162	Willing	− 6.8608 ± 3.7678	− 14.2457	0.5241	False
16	Feature 440	Teeth	− 6.8217 ± 3.8143	− 14.2977	0.6543	False
17	Feature 952	Listen	− 6.7891 ± 3.431	− 13.5139	− 0.0643	True
18	Feature 538	Brick	− 6.7806 ± 3.8367	− 14.3005	0.7393	False
19	Feature 17	Legs	− 6.7404 ± 4.1335	− 14.8421	1.3613	False
20	Feature 849	Kill	6.7367 ± 3.8827	− 0.8734	14.3468	False

Table 8 Comparison between top-20 word uni-gram and ELMo features (proposed by OVB-LR model)

S #	Top-20 ELMo features				Top-20 word uni-gram features		
	Name	Words	Weight	Salient	Name	Weight	Salient
1	Feature 361	Khan	− 10.0693 ± 3.8836	True	Listen	0.64 ± 0.1267	True
2	Feature 633	Pakistan	− 9.9991 ± 3.7884	True	Stop	0.4966 ± 0.0858	True
3	feature 532	Army	− 9.0647 ± 4.5108	True	Eyes	0.4457 ± 0.0892	True
4	Feature 1000	Anyone	− 8.5581 ± 3.8082	True	Kill	0.4335 ± 0.0861	True
5	Feature 343	Stupid	− 8.4728 ± 3.5904	True	Tear	0.4278 ± 0.0943	True
6	Feature 735	People	− 8.2892 ± 3.6098	True	Shut	0.4083 ± 0.0893	True
7	Feature 846	Eyes	− 8.1079 ± 3.596	True	Brick	0.3963 ± 0.1068	True
8	Feature 673	Imran	− 7.9851 ± 4.036	True	Disgraced	0.3755 ± 0.0955	True
9	Feature 60	Tear	7.8552 ± 4.1391	False	Khan	0.3732 ± 0.1001	True
10	Feature 606	Threaten	− 7.4519 ± 4.2112	False	Shoot	0.3715 ± 0.0884	True
11	Feature 371	Lesson	− 7.2345 ± 4.0077	False	Anyone	0.3712 ± 0.0925	True
12	Feature 941	Shut	− 6.9435 ± 4.1371	False	THREATENED	0.3683 ± 0.08	True
13	Feature 198	Stop	− 6.931 ± 3.7605	False	Lesson	0.3443 ± 0.1206	True
14	Feature 51	Burn	6.9124 ± 4.2313	False	Hang	0.3283 ± 0.1116	True
15	Feature 162	Willing	− 6.8608 ± 3.7678	False	Difficult	0.3272 ± 0.0763	True
16	Feature 440	Teeth	− 6.8217 ± 3.8143	False	Teeth	0.3259 ± 0.0901	True
17	Feature 952	Listen	− 6.7891 ± 3.431	True	Teach	0.3242 ± 0.119	True
18	Feature 538	Brick	− 6.7806 ± 3.8367	False	Break	0.3148 ± 0.0987	True
19	Feature 17	Legs	− 6.7404 ± 4.1335	False	Stupid	0.3014 ± 0.0966	True
20	Feature 849	Kill	6.7367 ± 3.8827	False	Legs	0.2967 ± 0.0877	True

Fig. 9 ELMo feature ranking proposed by SHAP method



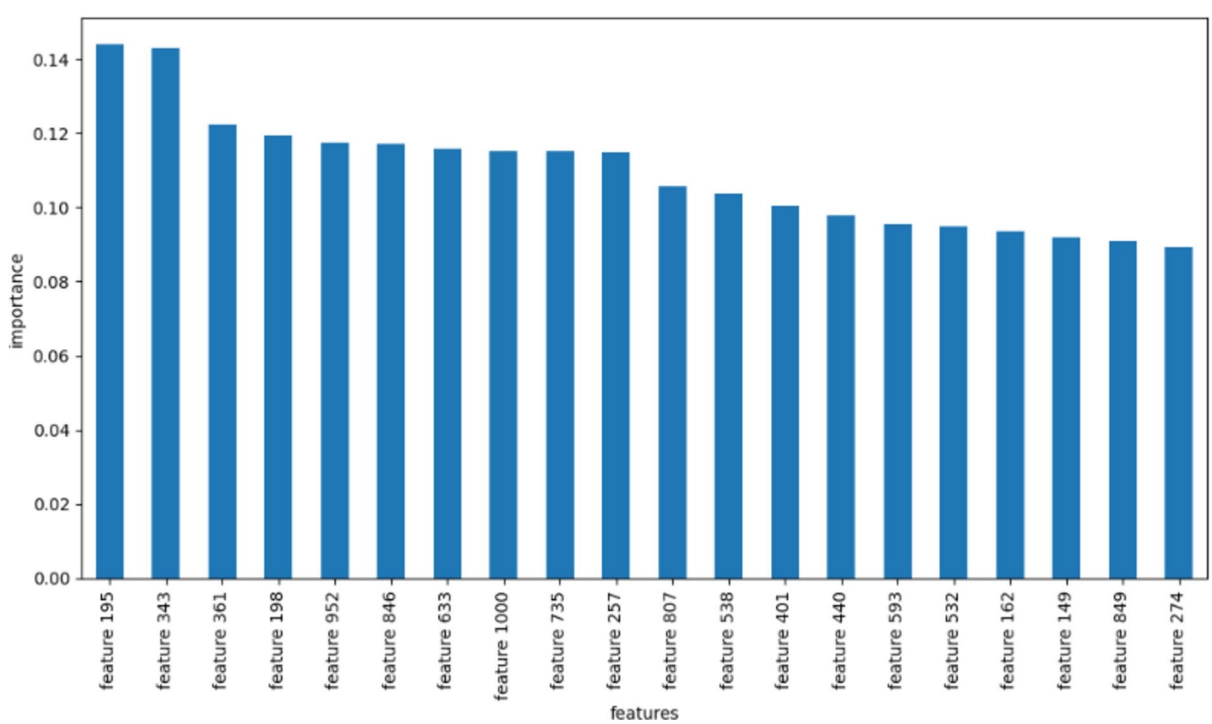
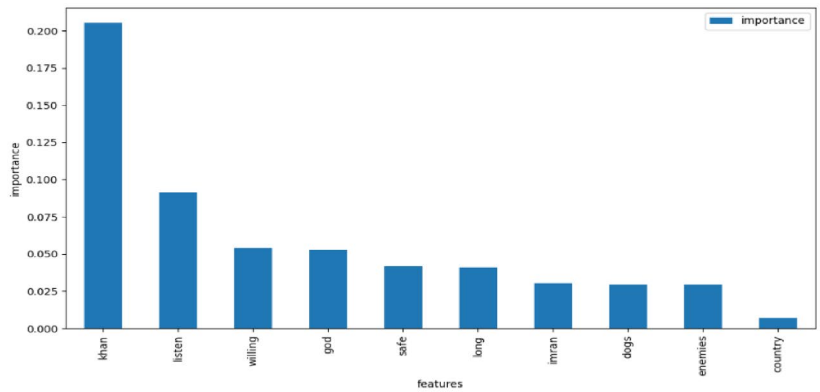


Fig. 10 Ranking proposed by AcME method for ELMo embedding features

Fig. 11 Interpretation of categorization of a threat comment using SHAP and AcME models

Tweet 1: Listen to Imran Khan and the enemies of our country. As long as Imran Khan is safe, everyone is safe. If anything happens to Imran Khan, even your dogs will not be spared. This is our promise, God willing.

a) SHAP force plot for Tweet 1 [Threat class identified], with base value and probability of class.



b) Weights proposed by the AcME Model to the words of Tweet 1 [threat class identified]

a serious impact on a victim's mental health and can be a sign of future harm. The United Kingdom, Brazil, Germany, and the United States are examples of countries that have laws that criminalize threats.

This study provides a detailed description of the process of identifying threat speech using state-of-the-art performance metrics, which can help prevent crime and serve as a basis for analyzing current social issues. The use of the interpretable model and algorithms helped to identify the main features that most influenced the assignment of an object to a threat and non-threat classes. This information also revealed the topics of interest to the users of these tweets. Currently, no other works have provided an interpretation of the results obtained through the use of XAI, in addition to classifying threat speech. To address this task, the OVB-LR model is proposed, which not only categorizes the tweets into threat or non-threat but also interprets the outcome. The model demonstrated benchmark performance on several metrics using ELMo embeddings and word uni-grams. The feature importance derived by the OVB-LR model is comparable with that of state-of-the-art SHAP and AcME models. The proposed methodology offers a reliable means of identifying threat content on social media and determining the features that distinguish between threat and non-threat classes.

Considering word uni-grams, the top-3 important features have no overlap between the three models. However, there are overlaps between pairs of models: SHAP and AcME both identify ‘*khan*’ as important (SHAP as 2nd, AcME as 1st), SHAP and OVB-LR both identify ‘*stop*’ as important (SHAP as 3rd, OVB-LR as 2nd), and there are no overlapping features between AcME and OVB-LR models. For ELMo embeddings, three models share one common feature in the top-3, i.e. ‘*feature 361*’. SHAP ranks this feature as 2nd, AcME as 3rd, and OVB-LR as 1st. There is an overlap between SHAP and AcME, with ‘*feature 195*’ being the most important for both models (ranked 1st for both). However, there are no overlaps between SHAP and OVB-LR, or between AcME and OVB-LR models.

Considering the baselines, the SVM model achieved the comparable f1-score for the threat class. This can be explained by the fact that it can handle non-linear separable data using kernel functions and can find the decision boundary that maximizes the margin between different classes, which is believed to improve the model's generalization performance. However, interpreting the results is more complicated due to the use of multiple spaces whereas the OVB-LR model produces better classification results and supports interpretable results. Likewise, the RF model showed considerable performance for threat class, macro, and weighted f1-scores. Even though this model can show feature importance, these metrics do not tell us for which class they play a major role during classification. In contrast, the OVB-LR model can determine which class a given feature plays a role in. We believe that the findings of this research will inspire the research community to explore threat comments classification and focus on the explainability of black-box models. The OVB-LR model can be used to moderate online content and challenge cases where the user disagrees with the model's verdict. This feature can reduce the number of online conflicts, false positive and false negative cases, making the moderation process more open and fair.

This research has some limitations. Firstly, the dataset was collected from Twitter, which has a character limit of 280 characters. Other platforms without character restrictions, such as YouTube, Facebook, or Reddit, could be used to overcome this limitation. Secondly, due to the dataset's size, the study's results cannot be generalized beyond the intended scope. Thirdly, the dataset used in this paper was obtained through translation. Despite manual editing, it lacks the special words and slang used by English-speaking Twitter users, which can be important indicators for identifying threatening comments. Additionally, the dataset was collected only from Pakistani Twitter accounts, further limiting its scope. Fourthly, the current work focused on binary classification. Further explorations can be made by extending the current framework to handle multi-class tasks such as threat speech can be divided into additional subclasses such as direct or indirect, personal or group. This will enable the model to address more diverse scenarios. Another direction is to add more visualization techniques for better explainability of the outcome of the model, ensuring deep insights and easier understanding.

6 Conclusion and future work

To the best of our knowledge, this is the first attempt in the field to obtain interpretable results for the threatening comment classification task. Additionally, a new balanced dataset for threatening speech in English has been constructed. We proposed an architecture for classifying threatening speech and interpreting the prediction using an inherently explainable approach. The study employed two feature extraction methods, including word uni-gram and ELMo embeddings. The classical machine learning models such as KNN, SVM, LR, and RF as well as SHAP and AcME post-hoc interpretable models are used as baselines. A new model (OVB-LR) with an inherited interpretability approach is utilized. Experiments have demonstrated that the OVB-LR model produced better results than classical ML models for classification tasks. Addressing features, with word uni-grams, the OVB-LR model achieved notable performance in accuracy (80.83%), macro, and weighted f1-scores (80.75% both). Specifically, for cases where ELMo is used, the OVB-LR model outperformed in all metrics, achieving the benchmark performance in accuracy (81.25%), f1-score for the threat class (80.85%), and non-threat class (81.63%), as well as for the macro and weighted f1-scores (81.24% both).

When interpreting the results, the OVB-LR model's explanations are comparable and better in some aspects to those of state-of-the-art SHAP and AcME models. The OVB-LR model introduced the idea of highlighting salient features whereas SHAP and AcME models support score-based importance of features. In addition, OVB-LR suggests the ranking by calculating the exact weight values "with lower and upper bounds of the 95% CI for each feature". The OVB-LR model ranked "listen, stop, eyes, kill, and tears" as top-5 features using word uni-grams. There are overlaps between important features suggested by OVB-LR, SHAP, and AcME models. However, the best classification performance is observed with ELMo features and insightful interpretability is observed with word uni-grams respectively.

For future work, several extensions can be considered. One direction is to explore the latest visualization techniques for better interpretability and explanations of the model's outcome, this will allow deep insights and better understanding. Another direction is to transform the supervised framework into semi-supervised and un-supervised models for threat comments identification with interpretability. This will lower the burden of labeled dataset construction as it is a time-consuming and manual activity. Another possibility is to explore deep learning-based XAI approaches for better explainability of the model's outcome and to improve performance.

Acknowledgements This article is the output of a research project implemented as part of the Basic Research Program at the National Research University Higher School of Economics (HSE University). Moreover, this research was supported in part by computational resources of HPC facilities at HSE University.

Author contributions Conceptualization: Muhammad Shahid Iqbal Malik; Methodology: Muhammad Shahid Iqbal Malik; Validation and Formal Analysis: Muhammad Shahid Iqbal Malik, Anna Nazarova; Data Curation: Muhammad Shahid Iqbal Malik, Anna Nazarova; Visualization: Muhammad Shahid Iqbal Malik, Anna Nazarova; Supervision: Muhammad Shahid Iqbal Malik; Writing—Original draft: Muhammad Shahid Iqbal Malik, Anna Nazarova; Writing—Review & Editing: Muhammad Shahid Iqbal Malik, Dmitry I. Ignatov, Ibrar Hussain

Funding No funding was received for conducting this study.

Data availability The generated dataset will be shared on request.

Declarations

Competing interests The authors have no competing interests to declare that are relevant to the content of this article.

References

Abbas Y, Malik MSI (2023) Defective products identification framework using online reviews. Electron Commer Res 23(2):899–920

Alda M (2021) Social media—statistics & facts. https://www.statista.com/topics/1164/social-networks/
Aldera S et al (2021) Exploratory data analysis and classification of a new arabic online extremism dataset. IEEE Access 9:161613–161626
Ali G, Malik MSI (2023) Rumour identification on Twitter as a function of novel textual and language-context features. Multimedia Tools Appl 82(5):7017–7038
Ali S et al (2023) Explainable artificial intelligence (XAI): what we know and what is left to attain trustworthy artificial intelligence. Inf Fusion 99:101805
Al-Shedivat M, Dubey A, Xing E (2020) Contextual explanation networks. J Mach Learn Res 21(194):1–44
Alvarez Melis D, Jaakkola T (2018) Towards robust interpretability with self-explaining neural networks. In: Advances in neural information processing systems, vol 31
Alvari H, Sarkar S, Shakarian P (2019) Detection of violent extremists in social media. In: 2019 2nd international conference on data intelligence and security (ICDIS). IEEE
Amjad M et al (2021) Threatening language detection and target identification in Urdu tweets. IEEE Access 9:128302–128313
Amodeo F et al (2022) OG-SGG: ontology-guided scene graph generation—a case study in transfer learning for telepresence robotics. IEEE Access 10:132564–132583
Angelino E et al (2018) Learning certifiably optimal rule lists for categorical data. J Mach Learn Res 18(234):1–78
Angelotti G, Díaz-Rodríguez N (2023) Towards a more efficient computation of individual attribute and policy contribution for post-hoc explanation of cooperative multi-agent systems using Myerson values. Knowl-Based Syst 260:110189
Ashraf N et al. (2020) Individual vs. group violent threats classification in online discussions. In: Companion proceedings of the web conference 2020
Azizan SA, Aziz IA (2017) Terrorism detection based on sentiment analysis using machine learning. J Eng Appl Sci 12(3):691–698
Bennetot A et al (2022) Greybox XAI: a neural-symbolic learning framework to produce interpretable predictions for image classification. Knowl-Based Syst 258:109947
Brendel W, Bethge M (2019) Approximating cnns with bag-of-local-features models works surprisingly well on imagenet. arXiv preprint arXiv:1904.00760
Chen C et al (2019) This looks like that: deep learning for interpretable image recognition. In: Advances in neural information processing systems, vol 32
Ciravegna G et al (2023) Logic explained networks. Artif Intell 314:103822
Dale D et al (2021) Text detoxification using large pre-trained neural models. arXiv preprint arXiv:2109.08914.
Dandolo D et al (2023) AcME—Accelerated model-agnostic explanations: fast whitening of the machine-learning black box. Expert Syst Appl 214:119115
Du M, Liu N, Hu X (2019) Techniques for interpretable machine learning. Commun ACM 63(1):68–77
Ferrara E et al (2016) Predicting online extremism, content adopters, and interaction reciprocity. In: Social informatics: 8th international conference, SocInfo 2016, Bellevue, WA, USA, 11–14 Nov 2016, Proceedings, Part II 8. Springer
Fortuna P, Soler J, Wanner L (2020) Toxic, hateful, offensive or abusive? what are we really classifying? an empirical analysis of hate speech datasets. In: Proceedings of the 12th language resources and evaluation conference
Ghaeini R et al (2019) Saliency learning: teaching the model where to pay attention. arXiv preprint arXiv:1902.08649
Gupta P, Varshney P, Bhatia M (2017) Identifying radical social media posts using machine learning. GitHub, California
Hammer HL et al. (2019) Threat: a large annotated corpus for detection of violent threats: In: 2019 International conference on content-based multimedia indexing (CBMI). IEEE
Hendricks LA et al (2016) Generating visual explanations. In: Computer vision–ECCV 2016: 14th European conference, Amsterdam, The Netherlands, 11–14 Oct 2016, Proceedings, Part IV 14. Springer
Hind M et al (2019) TED: teaching AI to explain its decisions. In: Proceedings of the 2019 AAAI/ACM conference on AI, Ethics, and Society
Hoang NP, Pishva D (2014) Anonymous communication and its importance in social networking. In: 16th international conference on advanced communication technology. IEEE
Hossain MS (2015) Social media and terrorism: threats and challenges to the modern era. South Asian Survey 22(2):136–155
Hussain S, Malik MSI, Masood N (2022) Identification of offensive language in Urdu using semantic and embedding models. PeerJ Comput Sci 8:e1169
Jaeger H (2014) Controlling recurrent neural networks by conceptors. arXiv preprint arXiv:1403.3369
Jung J et al (2017) Simple rules for complex decisions. arXiv preprint arXiv:1702.04690
Kaati L et al (2015) Detecting multipliers of jihadism on twitter. In: 2015 IEEE international conference on data mining workshop (ICDMW). IEEE
Kaczmarek-Majer K et al (2022) PLENARY: explaining black-box models in natural language through fuzzy linguistic summaries. Inf Sci 614:374–399
Khan MS, Malik MSI, Nadeem A (2024) Detection of violence incitation expressions in Urdu tweets using convolutional neural network. Expert Syst Appl 245:123174
Lakkaraju H, Bach SH, Leskovec J (2016) Interpretable decision sets: a joint framework for description and prediction. In: Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining
Liu J et al (2024) Small samples-oriented intrinsically explainable machine learning using Variational Bayesian Logistic Regression: an intensive care unit readmission prediction case for liver transplantation patients. Expert Syst Appl 235:121138
Liu H, Wang W, Li H (2022) Towards multi-modal sarcasm detection via hierarchical congruity modeling with knowledge enhancement. arXiv preprint arXiv:2210.03501
Lundberg SM, Lee S-I (2017) A unified approach to interpreting model predictions. In: Advances in neural information processing systems, 30
Malik MSI (2023) Threatening expression and target identification in under-resource languages using NLP techniques. In: International conference on analysis of images, social networks and texts. Springer
Malik MSI, Nawaz A (2024) SEHP: stacking-based ensemble learning on novel features for review helpfulness prediction. Knowl Inf Syst 66(1):653–679
Malik MSI, Cheema U, Ignatov DI (2023a) Contextual embeddings based on fine-tuned Urdu-BERT for Urdu threatening content and target identification. J King Saud Univ-Comput Inf Sci 35(7):101606
Malik MSI, Imran T, Mamdouh JM (2023b) How to detect propaganda from social media? Exploitation of semantic and fine-tuned language models. PeerJ Comput Sci 9:e1248
Malik MSI et al (2023c) Multilingual hope speech detection: a Robust framework using transfer learning of fine-tuning RoBERTa model. J King Saud University-Comput Inf Sci 35(8):101736
Malik MSI et al (2024a) Categorization of tweets for damages: infrastructure and human damage assessment using fine-tuned BERT model. PeerJ Comput Sci 10:e1859

Malik MSI, Rehman F, Ignatov DI (2024b) Ensemble learning with linguistic, summary language and psychological features for location prediction. Int J Inf Technol 16(1):193–205
Malik MSI et al (2023) Effectiveness of ELMo embeddings, and semantic models in predicting review helpfulness. Intell Data Anal 1–21 **(preprint)**
Mao R et al (2022) The biases of pre-trained language models: an empirical study on prompt-based sentiment analysis and emotion detection. IEEE Trans Affect Comput 14(3):1743–1753
Mehboob A, Malik M (2021) Smart fraud detection framework for job recruitments. Arab J Sci Eng 46(4):3067–3078
Mussiraliyeva S et al (2020) On detecting online radicalization and extremism using natural language processing. In 2020 21st International Arab conference on information technology (ACIT). IEEE
Nawaz A, Malik MSI (2022) Rising stars prediction in reviewer network. Electron Commer Res 22(1):53–75
Nouh M, Nurse JR, Goldsmith M (2019) Understanding the radical mind: Identifying signals to detect extremist content on twitter. In: 2019 IEEE international conference on intelligence and security informatics (ISI). IEEE
Papernot N, McDaniel P (2018) Deep k-nearest neighbors: Towards confident, interpretable and robust deep learning. arXiv preprint arXiv:1803.04765
Park DH et al (2018) Multimodal explanations: justifying decisions and pointing to the evidence. In: Proceedings of the IEEE conference on computer vision and pattern recognition
Rehan M, Malik MSI, Jamjoom MM (2023) Fine-tuning transformer models using transfer learning for multilingual threatening text identification. IEEE Access
Ribeiro MT, Singh S, Guestrin C (2016) Why should i trust you?" Explaining the predictions of any classifier. in Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining
Rudin C et al (2022) Interpretable machine learning: fundamental principles and 10 grand challenges. Stat Surv 16:1–85
Rudin C et al (2022) Interpretable machine learning: fundamental principles and 10 grand challenges. Stat Surv 16(1):1–85. https://doi.org/10.1214/21-SS133
Saarela M, Jauhiainen S (2021) Comparison of feature importance measures as explanations for classification models. SN Appl Sci 3(2):272
Saisubramanian S, Galhotra S, Zilberstein S (2020) Balancing the tradeoff between clustering value and interpretability. In: Proceedings of the AAAI/ACM conference on AI, ethics, and society
Scanlon JR, Gerber MS (2015) Forecasting violent extremist cyber recruitment. IEEE Trans Inf Forensics Secur 10(11):2461–2470
Schmitz GP, Aldrich C, Gouws FS (1999) ANN-DT: an algorithm for extraction of decision trees from artificial neural networks. IEEE Trans Neural Netw 10(6):1392–1401
Sharif W et al (2019) An empirical approach for extreme behavior identification through tweets using machine learning. Appl Sci 9(18):3723
Somerville K (2011) Violence, hate speech and inflammatory broadcasting in Kenya: the problems of definition and identification. Ecquid Novi Afr J Stud 32(1):82–101
Ustun B, Rudin C (2016) Supersparse linear integer models for optimized medical scoring systems. Mach Learn 102:349–391
Viljoen F (2005) Inciting violence and propagating hate through the media: Rwanda and the limits of international criminal law. Obiter 26(1):26–40
Widmer CL et al (2023) Towards human-compatible XAI: explaining data differentials with concept induction over background knowledge. J Web Semant 79:100807
Wu M et al (2018) Beyond sparsity: tree regularization of deep models for interpretability. In: Proceedings of the AAAI conference on artificial intelligence
Xiao L et al (2022) Exploring fine-grained syntactic information for aspect-based sentiment classification with dual graph neural networks. Neurocomputing 471:48–59
Xiao L et al (2024) Atlantis: Aesthetic-oriented multiple granularities fusion network for joint multimodal aspect-based sentiment analysis. Inf Fusion 106:102304
Younas MZ, Malik MSI, Ignatov DI (2023) Automated defect identification for cell phones using language context, linguistic and smoke-word models. Expert Syst Appl 227:120236
Zhang Q, Wu YN, Zhu S-C (2018) Interpretable convolutional neural networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition