PeerJ Computer Science

Beyond words: a hybrid transformer-ensemble approach for detecting hate speech and offensive language on social media

Uzair Iftikhar[1], Syed Farooq Ali[1], Ghulam Mustafa[1], Nurhidayah Bahar[2] and Kashif Ishaq[1]

[1] School of Systems and Technology, University of Management & Technology, Lahore, Lahore, Pakistan
[2] Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia, Bangi, Selangor, Malaysia

ABSTRACT

Over the past decade, there has been an increase in hateful content on social media platforms, specifically in tweets. Hence, it brings a challenge to identify and classify tweets containing racism, discrimination, offense, toxicity, or abuse. This article proposes a novel hybrid approach that combines the power of Transformer-based language modeling with ensemble learning to classify offensive, toxic, and hateful tweets. Specifically, the robustly optimized bidirectional encoder representations from Transformers pretraining approach (RoBERTa)-large model is employed for feature extraction, followed by a hyperparameter-tuned extreme gradient boosting (XGBoost) classifier for final classification. The approach was evaluated on three widely used datasets ToxicTweets, Davidson, and HateSpeechDetection and compared against state-of-the-art methods, including deep architectures such as convolutional neural network (CNN), bidirectional encoder representations from Transformers (BERT), and AngryBERT; transformer models including DistilBERT, RoBERTa, A Lite BERT (ALBERT), and efficiently learning an encoder that classifies token replacements accurately (ELECTRA); and logistic regression. The experimental results demonstrate that the proposed hybrid model significantly outperforms existing approaches in terms of accuracy, precision, recall, and F1-score, achieving the highest accuracy of 97% on the HateSpeechDetection dataset and 92.42% on the Davidson dataset. Furthermore, the approach was compared with other ensemble methods, including Adaptive Boosting (AdaBoost), random forest, support vector classifier (SVC), light gradient boosting machine (LightGBM), and bagging, to highlight its superior performance. This study suggests that integrating RoBERTa-large with XGBoost is an effective approach for hate speech detection and provides a robust solution to the increasing problem of online hate and toxicity.

Subjects Algorithms and Analysis of Algorithms, Artificial Intelligence, Data Mining and Machine Learning, Natural Language and Speech, Sentiment Analysis
Keywords Hate speech, Offensive language, Sentiment analysis, Social media, Text classification

Submitted 27 January 2025
Accepted 22 August 2025
Published 16 October 2025

Corresponding authors
Nurhidayah Bahar,
nbahar@ukm.edu.my
Kashif Ishaq,
kashif.ishaq@umt.edu.pk

Academic editor
Luigi Di Biasi

Additional Information and Declarations can be found on page 26

DOI **10.7717/peerj-cs.3214**

How to cite this article Iftikhar U, Ali SF, Mustafa G, Bahar N, Ishaq K. 2025. Beyond words: a hybrid transformer-ensemble approach for detecting hate speech and offensive language on social media. **PeerJ Comput. Sci. 11:e3214 DOI 10.7717/peerj-cs.3214**

INTRODUCTION

In the modern era, a large portion of the world's population interacts with social networking sites such as Facebook, YouTube, Twitter, Instagram, and Reddit to spread and exchange ideas. These digital venues give users the freedom to express themselves in a variety of ways, including blogs, online forums, and professional networking websites, both publicly and anonymously. In the daily generation of massive amounts of user-driven content, some individuals or groups are using these platforms to spread hate speech. Such misuse can have devastating effects on target audiences as it encourages and distributes harmful sentiments. Although digital communication enables people of diverse backgrounds to share ideas and opinions, it is important to recognize that these virtual spaces can also be sites of harassment and verbal abuse. Such environments can have many negative consequences, including the proliferation of hate speech, the use of demeaning language, cyberbullying, racist comments, discrimination based on gender, and several other forms of harassment.

Hate speech includes statements designed to intimidate, degrade, or ridicule a person or group based on characteristics such as skin color, national origin, gender, race, religious beliefs, political preferences, *etc.* It may also arise from affiliation with certain groups, provoking individuals, or enacting discrimination against particular persons or communities (*Pereira-Kohatsu et al., 2019*). According to a study by *Fortuna & Nunes (2018)*, hate speech is recognized as a verbal communication that marginalizes or assaults specific groups, which can incite violence or hatred against them because of distinguishing characteristics such as physical appearance, religious affiliation, gender, race, or nationality. This kind of speech may be conveyed in various tones, sometimes even accompanied by perverse humor. Offensive language, on the other hand, includes comments or posts that are disrespectful, demeaning, or bullying toward others, including abusive or sarcastic terms.

The pervasiveness of online hate has a profound impact on society and individuals. In 2021, the Anti-Defamation League (ADL), an organization at the forefront of combating hate, conducted a study to measure the extent of hate and abuse on the Internet (*Anti-Defamation League, 2025*). Findings from the ADL study (*Anti-Defamation League, 2025*) showed that the consequences of online hate are profound, with nearly 23% of individuals reporting sleep disturbances and 11% with suicidal thoughts, highlighting a disturbing trend that affects social well-being.

Figure 1 presents a breakdown of the different ways in which online hate and harassment affect individuals, as identified in the ADL survey. Therefore, identifying and eliminating hate speech is essential for creating a safe and more welcoming online environment. In the age of big data, with infinite amounts of information published daily, manually filtering and categorizing all data is a tedious and inefficient task (*Mullah & Zainon, 2021*). Human limitations, such as fatigue and varying levels of skill, can affect the accuracy of manual classification. Thus, leveraging machine learning (ML) methods in classification workflows offers a more efficient and unbiased approach. ML algorithms can

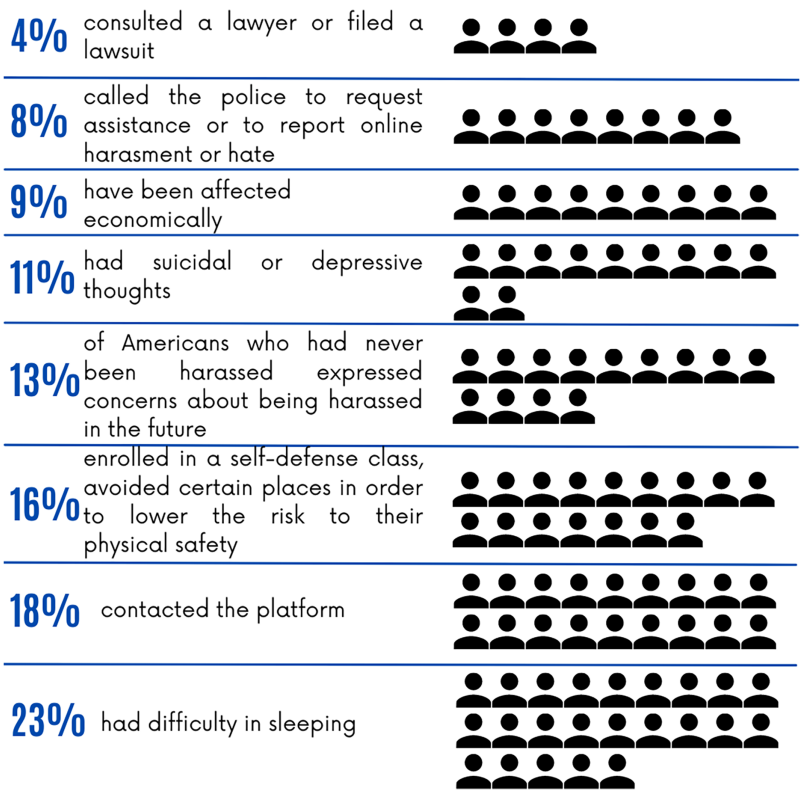

Figure 1 Impacts of online hate and harassment on individuals according to ADL's survey (*Anti-Defamation League, 2025*). Full-size DOI: 10.7717/peerj-cs.3214/fig-1

streamline the process, making it easier to quickly and accurately determine whether the content is hateful or not.

For our research, we chose Twitter as a focal point because of its reputation as one of the most controversial social media networks. A survey, conducted by SimpleTexting in March 2022, gathered insights from 1,018 social media users aged 18 to 75 years in the United States, revealed that participants perceive Twitter as particularly toxic (*Norton, 2022*). The results of this survey, shown in Fig. 2, employ a scale from 1 to 10 to quantify toxicity, with 1 indicating minimal toxicity and 10 indicating extreme toxicity. Based on the survey, Twitter's toxicity was measured at 7.82, making it the most hateful and toxic platform. The survey compared it with other platforms, as shown in Fig. 2, where Reddit scored 7.63 in toxicity, Facebook 7.47, and TikTok 6.83. Twitter is also the most widely used microblogging social media platform (*Singh, 2025*), with nearly 450 million monthly active users as of 2023 (*Ruby, 2023*). Almost 500 million tweets are seen daily on Twitter in which individuals share their thoughts and feelings about each other, with English being the primary language used.

For our research, datasets composed of English-language tweets were utilized. Figure 3 shows the distribution of Twitter users by country, with the United States leading with about 77.75 million users—where English is the dominant language—Japan follows with about 58.2 million users, and India with an estimated 24.45 million Twitter users (*Shepard,*

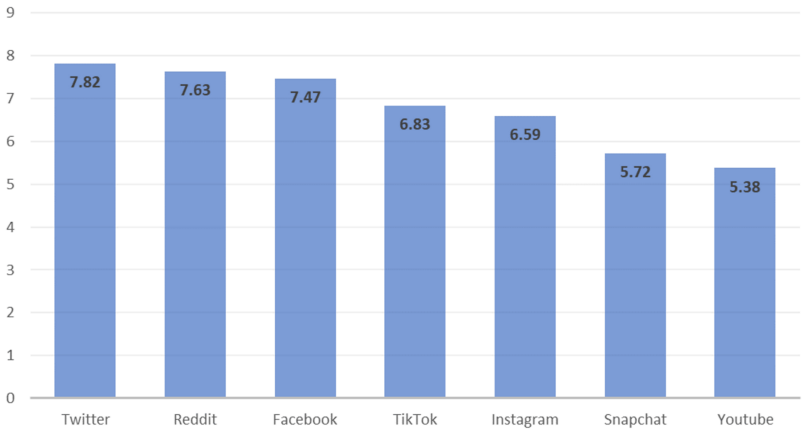

Figure 2 **Ratings of users regarding the toxicity of social media apps on a scale of 1–10, where 1 is the lowest and 10 is the highest (*Norton, 2022*).** Full-size DOI: 10.7717/peerj-cs.3214/fig-2

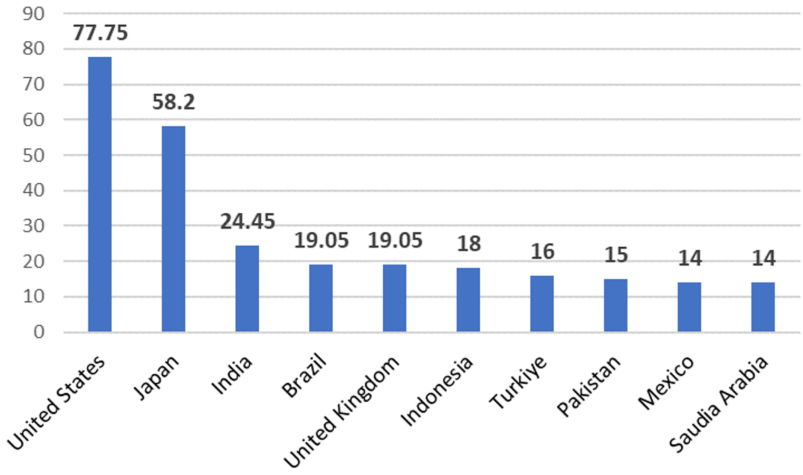

Figure 3 **Number of Twitter users by country in millions (*Shepard, 2023*).** Full-size DOI: 10.7717/peerj-cs.3214/fig-3

2023). This study proposes a novel approach based on a hybrid transformer-ensemble model to classify various types of hateful tweets, including toxic, offensive, and abusive tweets, using Davidson (*Davidson et al., 2017*), HateSpeechDetection (https://doi.org/10.6084/m9.figshare.19686954.v1) (HSD), and ToxicTweets (https://www.kaggle.com/datasets/ashwiniyer176/toxic-tweets-dataset) (TT) datasets.

The key contributions of this article are summarized as follows:

- This article introduces a novel hybrid approach that combines the RoBERTa-large model for feature extraction with a hyperparameter-tuned XGBoost classifier to effectively detect toxic, offensive, and abusive tweets.
- Our approach addresses key gaps in hate speech detection by eliminating manual feature engineering *via* robustly optimized bidirectional encoder representations from Transformers pretraining approach (RoBERTa) embeddings, reducing data dependency

through transfer learning, and optimizing hybrid integration with tuned transformer-ensemble synergy.

- The proposed approach (PA) achieves impressive accuracy scores of 95%, 92.42%, and 97% on the TT, Davidson, and HSD datasets, respectively.
- The PA significantly outperforms state-of-the-art methods, including Transformer-based models (DistilBERT, RoBERTa, A Lite BERT (ALBERT), and efficiently learning an encoder that classifies token replacements accurately (ELECTRA)), deep learning (DL) models (AngryBERT, bidirectional encoder representations from Transformers (BERT), convolutional neural network (CNN)), a natural language process (NLP)-based method (*Mosca, Wich & Groh, 2021*), and logistic regression, achieving an accuracy of 92.42% on Davidson dataset.
- The PA also surpasses these existing techniques in terms of precision, recall, and F1-score, with values of 91.03%, 92.33%, and 91.24%, respectively on Davidson dataset.
- The PA is compared against five alternative classifiers, including ensemble models such as random forest classifier (RFC), Bagging, AdaBoost, and light gradient boosting machine (LGBM), along with support vector classification (SVC). PA consistently achieves higher accuracy across all three datasets.
- Furthermore, PA demonstrates superior precision, recall, and F1-score compared to these alternative classifiers across all datasets, highlighting its robustness and generalizability.

The rest of this article is organized as follows. 'Literature Review' covers a summary of relevant articles, while 'Methodology' covers the proposed methodology. 'Experiment & Results' contains the experiment & results. 'Result Analysis' presents a detailed analysis and discussion of the results. 'Conclusion & Discussion' concludes the article.

LITERATURE REVIEW

Efforts to create a secure online community have led to the development of ML and DL models aimed at identifying and mitigating hate speech along with offensive content on social media. This section delves into these models and their underlying algorithms. Table 1 provides an overview of existing studies focused on the detection of various forms of abusive language. Researchers have explored a variety of feature extraction and feature selection methods to improve hate speech detection, incorporating techniques such as bag-of-words (*Kwok & Wang, 2013*; *Sharma, Agrawal & Shrivastava, 2018*; *Malmasi & Zampieri, 2017*), n-grams (*Nobata et al., 2016*; *Waseem & Hovy, 2016*; *Dinakar, Reichart & Lieberman, 2011*), deep learning methods (*Köffer et al., 2018*), TF-IDF vectorization (*Köffer et al., 2018*; *Liu & Forss, 2014*), and feature selection methods that improve sentiment classification, such as those combining rough set theory with optimization techniques (*Muhammad, Abdullah & Sani, 2021*). Some studies also focused on improving the computational efficiency of DL models using frameworks such as the Ktrain Library (*Othman & Yaakub, 2022*).

Research on automated hate speech detection has evolved through three key methodological paradigms, each addressing limitations of prior approaches while

Table 1 Summary of the key research articles in terms of Accuracy (Acc.), Precision (Prec.), Recall (Rec.), F1-score (F1-scr.).

Author	Dataset detail	Model	Features	Acc.	Prec.	Rec.	F1-scr.	Limitation
Abro et al. (2020)	14.5k tweets	SVM	Bigram with TF-IDF	0.79	0.77	0.79	0.77	Failed to assess message severity
Khanday et al. (2022)	11k COVID-19 tweets	SGB	TF-IDF; BOW	0.9804	0.99	0.97	0.98	Single dataset utilized
Alfina et al. (2017)	713 tweets	RFC; DT	Word n-gram	–	–	–	93.5%	Small dataset was used
Davidson et al. (2017)	Dvd (Hatebase) dataset of 24k tweets	LR with L2 regularization	Unigram, bigram, trigram with TF-IDF	0.89	0.91	0.90	0.90	Misclassification; limited evaluation metrics
Greevy & Smeaton (2004)	dataset from PRINCIP	SVM	BOW	0.9150	0.9278	0.90	0.9137	Single ML model employed
Gambäck & Sikdar (2017)	ZeerakW dataset of 6.7k tweets	CNN	word2vec	–	0.8566	0.7214	0.7829	Limited tweets; single deep model employed
Ishmam & Sharmin (2019)	2,000 comments from Facebook	GRU	word2vec	0.7010	0.68	0.70	0.69	Dataset with few instances
Mathew et al. (2021)	HateXplain 20k posts from Gab and Twitter	CNN-GRU, BiRNN, BERT	–	0.698	–	–	0.687	Lack of preprocessing
Biradar, Saumya & Chauhan (2022)	4,575 tweets	TIF-DNN	mBERT tokenizer	0.73	–	–	–	Mistranslation; misclassification
Pitsilis, Ramampiaro & Langseth (2018)	Wsm 16k tweets	RNN (LSTM)	Word-based frequency vectorization	–	0.9305	0.9334	0.9320	Only one dataset used
Ghosh, Ghosh & Das (2017)	882 posts	WEKA software	Word-based & semantic features	0.685	–	–	–	Dataset with few instances
Vidgen & Yasseri (2020)	4,000 tweets	SVM with radial kernel	gloVe word embeddings	0.773	0.778	0.773	0.776	Limited tweets; focused solely on "Islamophobia"
Mutanga, Naicker & Olugbara (2020)	Dvd 24k tweets	DistilBERT	–	0.92	0.75	0.75	0.75	Lack of preprocessing
Mozafari, Farahbakhsh & Crespi (2020)	Wsm 16k+6.9k & Dvd 24k tweets	BERT (based) + CNN	–	–	89% & 92%	87% & 92%	88% & 92%	High misclassification

introducing new challenges. In the following, we analyze prominent ML, DL and transformer-based (TL) techniques from the recent literature, highlighting their comparative advantages and unresolved limitations, which motivate our hybrid approach.

Machine learning based approaches

Early machine learning (ML) approaches demonstrated strong performance on explicit hate speech. *Greevy & Smeaton (2004)* tackled the issue of identifying racist text using supervised ML techniques. The research involved transforming textual data into numerical vectors through the extraction of bigram features, supplementary to other methods such as bag-of-words (BOW) and part-of-speech (POS) tagging. To analyze the outcomes of their experiments, they used the SVM classifier. They documented a high accuracy rate of 91.50% when employing the polynomial kernel function alongside BOW representations.

Building on these foundations, *Davidson et al. (2017)* employed a combination of unigrams, bigrams, and trigrams, each weighted by TF-IDF, to assess the content of tweets sourced *via* a crowdsourced hate speech lexicon. The tweets were divided into three categories (hate speech, offensive language, or neither), leveraging the crowdsourcing method for categorization. To handle the complexity of data, they utilized L1 and L2 regularization techniques and applied LR and support vector machines (SVM) to train their models. They obtained a precision, recall, and F1-score of 91%, 90%, and 90%, respectively.

More recently, *Khanday et al. (2022)* classified hate speech on Twitter during the COVID-19 era by employing TF-IDF, BOW, and Tweet length engineering techniques for feature extraction. The authors fed the generated data to different ML algorithms as well as applied different ensemble learning techniques. According to their findings, the SGB classifier exceeds all others with precision, recall, F1-score, and accuracy of 98.04%, 99%, and 97%, respectively. Although these ML methods achieve high accuracy, they depend on manual feature engineering and struggle with contextual nuances like sarcas, which require domain-specific tuning for optimal performance.

Deep learning based approaches

Later work leveraged the learned representations to reduce feature engineering. *Gambäck & Sikdar (2017)* explored a variety of methods for tweet classification, incorporating character 4-grams, semantic-based word vectors, random word vectors, and a combination of word vectors and character n-grams. They implemented max pooling to simplify the features before applying a softmax function for the final tweet classification. Their research concluded that the model incorporating word2vec embeddings was superior in terms of performance, achieving an F1-score of 78.3% validated by a 10-fold cross-validation process. Extending these techniques to multilingual contexts, *Ghosh, Ghosh & Das (2017)* carried out sentiment identification using code-mixed text data obtained from social media. The author experimented using English-Bengali and English-Hindi code-mixed datasets. Data were categorized as positive, negative, or neutral based on the polarity of the statement. They achieved a polarity classification accuracy of 68.5% by utilizing a multilayer perception model in conjunction with SentiWordNet, opinion lexicon, and POS tags.

For sequential modeling, *Pitsilis, Ramampiaro & Langseth (2018)* developed an ensemble method using recurrent neural networks (RNN) with LSTM architecture to analyze a dataset of roughly 16,000 publicly available tweets. They transformed tweets into

vector form through word frequency vectorization and included additional features to gauge the likelihood of users posting hate speech. To enhance the classifier's accuracy, the outputs from multiple LSTM classifiers were aggregated, resulting in a notable F1-score of 0.9320. *Ishmam & Sharmin (2019)* delved into the classification of a Bengali language comments dataset obtained from Facebook, using both ML algorithms and a deep neural network based on the gated recurrent unit (GRU) structure. The features were extracted using n-gram and character n-gram methodologies, each with their own TF-IDF values, after a series of data preprocessing steps. They fed the processed data to several ML algorithms and initially achieved an accuracy of 52.20%. This was further improved by using a GRU-based model, which increased the accuracy by approximately 18%, reaching 70%.

A significant research study in 2019 focused on creating a Danish dataset for the detection of hate speech and abusive language (*Sigurbergsson & Derczynski, 2019*). The dataset employed by the authors comprised comments from both Reddit and Facebook, along with details regarding the types and target audiences of offensive language. To achieve the highest F1-score of 0.74, the authors utilized DL models together with diverse feature sets. *Abro et al. (2020)* proposed an automatic scheme to detect hate speech messages. The author used various significant ML algorithms with different feature engineering techniques and compared the performance of the models. Their experimental results demonstrated that the SVM exhibited the highest accuracy of 79% when bigram features were applied.

Aljero & Dimililer (2021) proposed a novel approach to detect hate speech in English tweets by using a stacked ensemble architecture. Their method combined three classifiers, namely SVM, LR, and XGB trained on word2vec and universal encoding features. To generate the final output, they made the meta classifier LR to integrate the predictions from the base classifiers along with their corresponding features. The experimental outcomes of their proposed architecture demonstrated a performance improvement across the four datasets (HatEval, Davidson (Dvd), COVID-HATE, and ZeerakW) compared to standard stacking, base classifiers, and majority voting approaches. Although DL reduced feature engineering needs, these approaches still required large labeled datasets and struggled with computational costs, particularly for low-resource languages and real-time applications.

Transformer learning based approaches

Transformers revolutionized multilingual detection. *Biradar, Saumya & Chauhan (2022)* conducted a study in 2022 to detect hate speech in Hinglish. The study explored the efficacy of transformer models such as IndicBERT and multilingual Bidirectional Encoder Representation (mBERT), as well as transfer learning techniques utilizing pre-trained language models such as ULMFiT, and BERT. In addition, they proposed a deep neural network feature extraction model and a transformer-based interpreter (TIF-DNN). According to their experimental findings, their suggested model works better than current cutting-edge techniques for identifying hate speech in Hinglish with an accuracy of 73%.

Malik et al. (2024) conducted a comprehensive comparative study evaluating both traditional ML and DL based approaches for hate speech detection using three widely adopted benchmark datasets: Davidson, Founta, and Twitter Sentiment Analysis. Their work systematically assessed various embedding techniques, including TF-IDF, GloVe, and transformer-based models like BERT, Small BERT, ALBERT, and ELECTRA. Classifiers ranging from SVM and XGBoost to CNN and bidirectional long short-term memory (Bi-LSTM) were analyzed for effectiveness, efficiency, and domain generalization, providing a robust baseline for future transformer-ensemble integrations.

In the context of identifying hate speech on Twitter, text normalization has been underexplored, particularly its impact on handling lexical variants and enhancing model performance. *Mansur et al. (2024)* has proposed an improved normalization technique that combines rule-based patterns and the SymSpell spelling correction algorithm to convert out-of-vocabulary (OOV) words into recognized vocabulary, thus improving the effectiveness of sentiment and hate speech detection tasks. Although transformers reduce the dependency on text normalization, work like *Mansur et al. (2024)* demonstrates that careful OOV handling (*via* SymSpell+rules) can still boost performance by 12%, suggesting opportunities for hybrid approaches.

In a recent advancement, researchers explored the adaptation of five cutting-edge transformer models, ALBERT, DeBERTa, ELECTRA, HateBERT and DeepSeek, for the task of cyberbullying detection in social media contexts (*Philipo et al., 2025*). These models, originally developed for general text classification or sentiment analysis, were refined to address the nuanced and often implicit nature of harmful content online. The study highlighted how these modern architectures can effectively capture emotional tone and contextual cues, surpassing the limitations of traditional keyword-based methods. This underscores the growing importance of sentiment-aware transformer models in enhancing the sensitivity and contextual understanding required for robust hate speech and cyberbullying detection.

Research on online hate speech is hampered by limited resources, mainly affecting languages with limited data sets. These barriers are often due to the high cost and difficulty in collecting and managing training data in such cases. Recent advances in areas such as text (*Raffel et al., 2020*; *Brown et al., 2020*), image (*Ramesh et al., 2022*; *Saharia et al., 2022*), and speech generation (*Shen et al., 2018*) have opened up avenues for new analytical approaches. This enhancement enables the generation of synthetic data from model output, which in turn can be used to retrain new models. Data augmentation methods demonstrated their effectiveness in building a hate speech classifier (*Khullar et al., 2024*; *Ansari, Kaur & Saxena, 2024*; *Githa et al., 2024*), and demonstrated impressive performance. Additionally, some studies have addressed resource scarcity by developing sentiment corpora in low-resource languages, such as Malay, using semi-supervised learning methods (*Sukawai & Omar, 2020*).

Although multilingual studies have expanded to languages like German (*Velankar, Patil & Joshi, 2022*), Dutch (*Markov, Gevers & Daelemans, 2022*), and English (*del Valle-Cano et al., 2023*), most solutions face a trilemma: (1) ML/DL models struggle with contextual

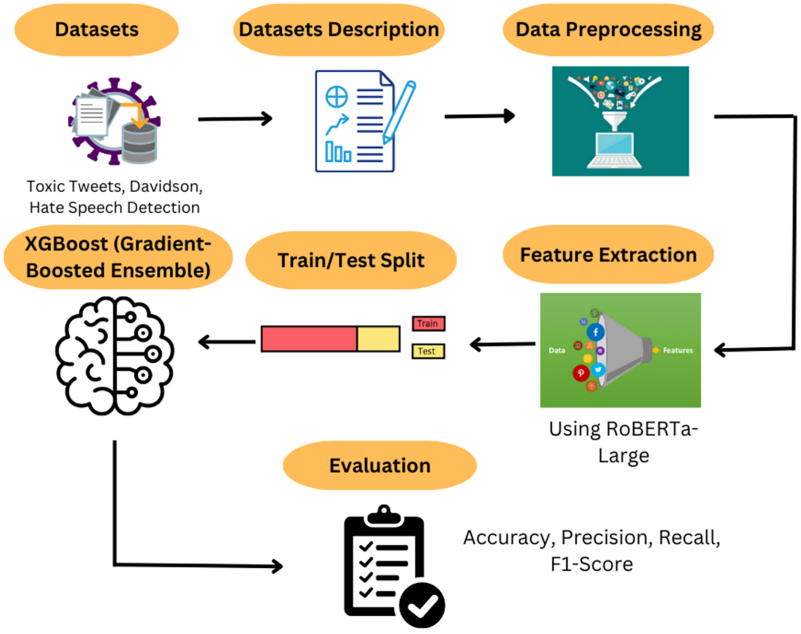

Figure 4 **Flow of proposed work.** Full-size DOI: 10.7717/peerj-cs.3214/fig-4

nuances, (2) pure transformer approaches overlook class imbalance, and (3) hybrid methods lack systematic optimization of transformer-ensemble synergy.

Our work addresses these gaps by introducing a novel hybrid architecture that combines RoBERTa-large's contextual depth with XGBoost's imbalance robustness, advancing beyond existing hybrid approaches (*Biradar, Saumya & Chauhan, 2022*) through rigorous hyperparameter tuning often overlooked in prior ensemble studies (*Aljero & Dimililer, 2021*). The proposed model demonstrates practical viability by achieving state-of-the-art performance across benchmark datasets (Davidson, HateSpeechDetection, ToxicTweets) while avoiding the extensive data requirements characteristic of pure DL methods (*Pitsilis, Ramampiaro & Langseth, 2018*), offering an optimized balance between transformer-based contextual understanding and ensemble learning's classification robustness.

METHODOLOGY

In this section, we discuss the methodology employed in our study. A Kaggle dataset is used as the foundation for developing a robust model to detect hate speech and offensive language. Additionally, we evaluated our PA on two other publicly available datasets. Our proposed model combines the power of a pretrained RoBERTa-Large model with an XGBoost classifier, enabling more accurate detection than traditional ML methods. The flow of our proposed work is illustrated in Fig. 4, which involves dataset description and preprocessing, feature extraction *via* RoBERTa-Large embeddings, a train-test split, and classification using XGBoost. Finally, we evaluate our model using various metrics and compare it with the baseline ML algorithms.

Table 2 Description of datasets.

Dataset	No. of tweets	Class label
TT	56,744	Non-Toxic = 0
		Toxic = 1
Davidson	25,296	Hateful = 0
		Offensive = 1
		Neither = 2
HSD	3,001	Not-Hateful = 0
		Hateful = 1

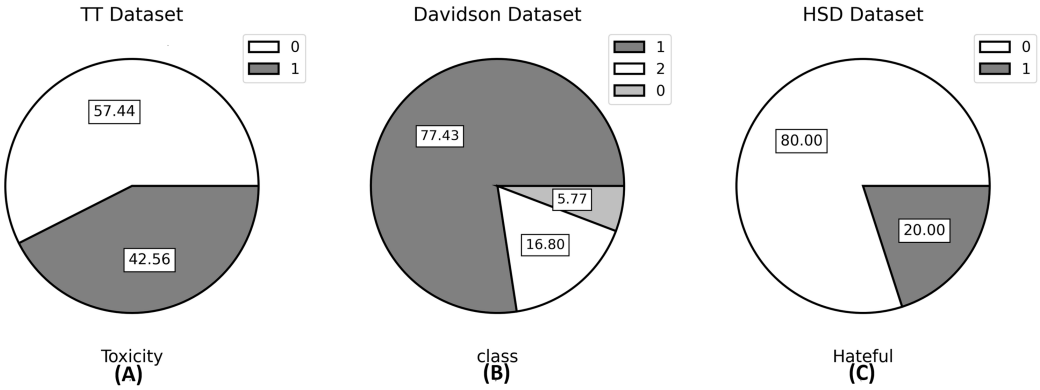

Figure 5 **Pie plots of the value count of the target variable for all three datasets.** In (A) Non-Toxic = 0; Toxic = 1, in (B) Hateful = 0; Offensive = 1; Neither = 2, and in (C) Not Hateful = 0; Hateful = 1.
Full-size DOI: 10.7717/peerj-cs.3214/fig-5

Datasets

Kaggle is one of the largest data source providers for learning purposes, so for this study we used a publicly available dataset of toxic tweets from Kaggle, namely the ToxicTweets (TT) dataset (*Iyer, 2021*). We prepare it appropriately to achieve our objectives. The dataset approximately includes 56,744 tweets. The dataset has a target variable, namely toxicity. We also evaluated our model on two other datasets that are also publicly available. One dataset is from literature, namely the Davidson dataset (*Davidson et al., 2017*) and the other is a HateSpeechDetection (HSD) (https://doi.org/10.6084/m9.figshare.19686954.v1) dataset (*Cooke, 2022*) which is a combination of comments from different platforms such as Twitter, Reddit, 4Chan, *etc*. The description of the datasets is given in Table 2.

The TT dataset has two classes, namely toxic which is labeled as 1 and non-toxic which is labeled as 0. Davidson's dataset has three classes; hateful, offensive, and neither; while the HSD dataset also has two classes; hateful and not hateful labeled 1 and 0, respectively. As far as our knowledge extends, the TT and HSD datasets are novel contributions to the field of hate speech detection, since no prior studies have been conducted on these two datasets. Therefore, our study introduces these datasets to the field for the first time. The visualization of datasets through a pie chart to see the percentage of each class is given in Fig. 5.

Table 3 Examples of some tweets from datasets.

Tweets	Nature
I was sleep by 10 pm last night, I feel great this morning.	Not hateful
KILL YOURSELF!!!	Hateful
He is one ugly pos	Hateful
In all fairness: some just call them n***ers	Hateful
Well that was depressing.	Not hateful

In previous research, the main issue researchers faced was labeling of the data, specifically determining which text is hateful and which is not. A significant challenge in this task is the difficulty in identifying hate speech, especially when it is disguised through sarcasm or lacks explicit words associated with hatred, stereotyping, or racism (*Mansur, Omar & Tiun, 2023*). The main cause of this issue is that a proper definition of hate speech was not available at that time. Over time, many researchers have proposed definitions of hate speech, which serve as the basis for labeling datasets today. Although the datasets used in this study were pre-labeled, we present examples of tweets from these datasets in Table 3 to illustrate instances of both hateful and non-hateful content. The presence of specific keywords in the text often serves as the basis for categorizing tweets. Table 3 helps provide a clearer understanding of the nature and structure of the data.

Preprocessing

Data manipulation or deletion before usage can be referred to as data preprocessing, which is done to ensure or improve performance. In this section, we discuss the steps that we used to clean the data to ensure that it performs well when a model is evaluated on it and generates better results.

- **Handling the diacritics in the text:** To handle the diacritics in the text, we used Python's "unicodedata" library, which allows standard character encoding. As we know, text in English is not readable for computers, so encoding using Unicode is necessary.
- **Replacement of Twitter handles and URLs:** We replaced all Twitter handles (*i.e.*, usernames starting with '@') and URLs (*i.e.*, any strings starting with 'http' or 'https') in tweets with an empty string.
- **Emoji removal:** We removed emojis from the text, as they do not contribute to the model's understanding of hate speech or offensive language.
- **Contractions expansion:** To preserve the full meaning of contractions (*e.g.*, "don't" to "do not"), we used the Python 'contractions' library to expand them, as expanded text improves interpretability for the model.
- **Lowercasing text:** All text was converted to lowercase to ensure consistency, as models often benefit from treating words in a case-insensitive manner.
- **Tokenization and Lemmatization:** We used the 'WordNetLemmatizer' from NLTK to lemmatize words after tokenizing the tweets. Lemmatization reduces

Algorithm 1 **Feature extraction pipeline using RoBERTa-Large.**

1. **Input:** Preprocessed text sequences $\{x_1, x_2, \ldots, x_n\}$ (after cleaning, lemmatization, *etc.*)
2. **Tokenization:** For each sequence x_i:
   Tokenize using RoBERTa-Large's tokenizer with:
   - Maximum sequence length $L = 130$
   - Apply dynamic padding to `max_length`
   - Create attention masks to ignore padding tokens
   - Apply truncation for sequences longer than L
3. **Model Inference:**
   - Pass tokenized inputs through RoBERTa-Large to obtain hidden states:
     `outputs = model.roberta(input_ids, attention_mask)`
   - Extract the last hidden state tensor of shape $[batch\_size, seq\_len, hidden\_dim]$
4. **Feature Pooling:**
   - Apply mean pooling along the sequence dimension to obtain fixed-size sentence embeddings:
     $f_i = \frac{1}{L}\sum_{j=1}^{L} h_{i,j}$
     where $h_{i,j}$ is the hidden state for token j in sequence i
5. **Output:**
   - Matrix of sentence embeddings $F \in \mathbb{R}^{n \times d}$ where $d = 1024$ (hidden size of RoBERTa-Large)
   - Use F as input features for the downstream XGBoost classifier

words to their base or root form, making the data more consistent and reducing redundancy.

- **Elimination of stop words:** Stop words, which are frequently used words with little semantic meaning, were removed to make training more efficient. Common words such as "and," "the," "is," *etc.*, were removed using NLTK's stop word list.
- **Removal of special characters, digits, and short words:** We removed all non-alphabetic characters, digits, and words with a length of 1 or 2 characters (such as "&", "#", "1", "4") as they do not contribute meaningfully to the detection of hate speech.
- **Concatenation of words:** Finally, the tokens of each tweet were concatenated into a single string, with each word separated by spaces. This is useful for further processing or analysis that requires data in a text string format.

Feature extraction using RoBERTa-Large

In this study, we used the pre-trained RoBERTa-Large model for feature extraction. RoBERTa is a transformer-based model that has shown substantial improvements in various NLP tasks due to its ability to generate contextualized embeddings. For traditional ML models, we employed TF-IDF as the feature extraction method, which represents text based on term frequency and inverse document frequency. Instead of relying on these frequency-based features, RoBERTa captures complex linguistic patterns by encoding each word in relation to the context of surrounding words.

During training, each tweet is passed through the RoBERTa-Large model to generate high-dimensional embeddings, which are then pooled to create a single feature representation for each text. This pooled embedding serves as the input feature for our classifier, significantly enhancing the model's ability to understand nuanced language,

Table 4 Distribution of data in training and testing sets for three datasets, with an 80/20 split.

Dataset	Total	Training set	Test set
TT	56,744	45,394	11,350
Davidson	25,296	20,236	5,060
HSD	3,001	2,401	600

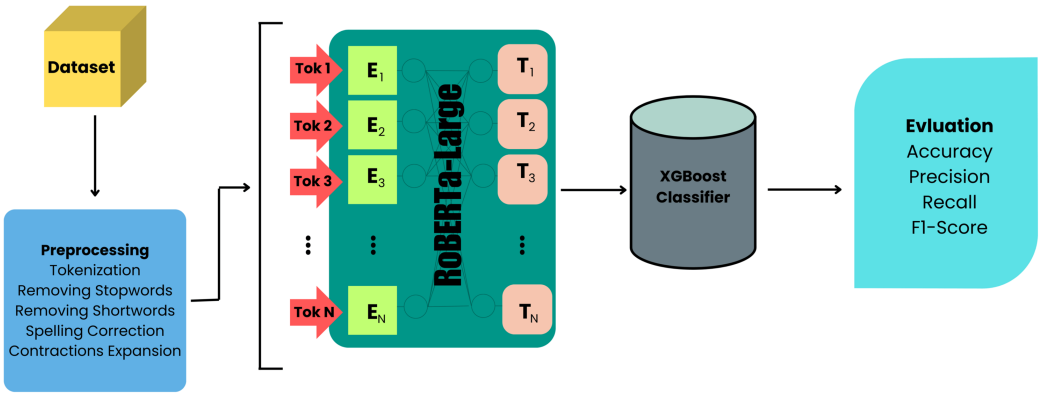

Figure 6 Proposed architecture diagram. Full-size DOI: 10.7717/peerj-cs.3214/fig-6

including hate speech and offensive language. The complete feature extraction pipeline is detailed in Algorithm 1.

Furthermore, data were segregated, with 80% designated for the training set and the remaining 20% for the testing set. Table 4 presents the distribution of data among the training and testing sets for all three datasets. This split is fundamental for a thorough assessment of the models, allowing for confident and trustworthy findings from both trained and novel test data.

Algorithmic description

ML is a field of study that focuses on developing algorithms that enable computers to learn from observations and interactions in the real world. These algorithms allow machines to simulate human-like behavior and enhance their ability to learn and perform well with input data (*Madaan et al., 2021*). ML employs a range of techniques to deal with data issues (*Mahesh, 2020*). Data scientists typically emphasize that there is no single "best" type of algorithm that is suitable for all situations. The choice of algorithm depends on various factors, such as the problem being addressed, the number of variables involved, and the type of model that will yield the best results. Other factors, such as available data and computing resources, also play a role in selecting the most appropriate algorithm for a given task.

In this study, we evaluated several ML algorithms in comparison to our proposed model. The following subsections provide an overview of each algorithm, including our new approach, RoBERTa-Large + XGBoost, which significantly improves the detection of hate speech and offensive language. Furthermore, Fig. 6 illustrates the key components and

their interconnections in the proposed algorithm/model, providing a visual representation of the methodology employed in this study.

Random forest classifier

The Random Forest algorithm, introduced by *Breiman (2001)*, is a method of ensemble learning that involves combining multiple decision trees to make a prediction. Each tree provides a classification for new data by randomly selecting features at each node. RF uses bagging to construct decision trees in the forest by sampling with replacement from the training set. The final prediction is made by averaging the probabilities determined by all trees. RF is effective when there are many variables compared to samples.

Let U be the input feature vector of size n, and let V be the corresponding output variable. Let D be a dataset of size $N = (u_1, v_1), (u_2, v_2), \ldots, (u_n, v_n)$, where u_i is an input vector of size n, and v_i is the corresponding output variable. Let T be the number of decision trees in the forest, and let M be the number of randomly selected features at each node. The random forest classifier (RFC) combines the predictions of multiple decision trees using Eq. (1).

$$P(V = 1|U = u) = \frac{1}{T}\sum_{t=1}^{T} P_t(V = 1|U = u) \tag{1}$$

where $P_t(V = 1|U = u)$ is the probability of the positive class for the input vector x predicted by the t-th decision tree. Each decision tree is constructed by randomly selecting a subset of M features at each node and training on a bootstrapped sample of the dataset. The final prediction is obtained by averaging the probabilities of all trees in the forest.

AdaBoost classifier

In machine learning, boosting is a technique that combines a number of relatively ineffective and poor prediction rules to produce a high accuracy rule (*Washburn et al., 2020*). A linear classifier, AdaBoost is a popular ensemble learning method that combines a number of weak classifiers to produce a more accurate classification model. The key idea behind AdaBoost is to iteratively train a sequence of weak learners and give more weight to the misclassified examples in each iteration (*Rawat & Suryakant, 2019*). The final model is obtained by combining the predictions of all weak learners with their assigned weights. AdaBoost classifiers can speed up processing while increasing classification accuracy. The AdaBoost algorithm can be expressed using Eq. (2).

$$F(x) = \sum_{n=1}^{N} \alpha_n f_n(x). \tag{2}$$

Here, $F(x)$ is the final prediction rule, α_n is the weight assigned to the n-th weak learner, $f_n(x)$ is the prediction made by the n-th weak learner on input x, and N is the total number of weak learners used in the ensemble.

Bagging classifier

The bagging classifier is a type of ensemble meta-estimator used in machine learning. It works by fitting base classifiers, such as decision trees, to random subsets of the original

dataset. The final prediction is generated by aggregating the individual predictions of each base classifier through voting or averaging. This randomization helps reduce the variance of the estimator, which can lead to more accurate predictions. This algorithm incorporates various works of literature. When a sample is drawn with replacement from the dataset, this method is called Bagging (*Breiman, 1996*).

Let X be a dataset of size N, and let X_i be a random subset of X of size n, where $i = 1, 2, \ldots, m$ is the number of base classifiers. Let $H(x; X_i)$ be a base classifier that produces a binary output for an input x, and let $y_i(x)$ be the output of the i-th base classifier on input x. The Bagging classifier combines the predictions of the base classifiers using majority voting, which is defined in Eq. (3).

$$f(x) = argmax\ i \sum j = 1^m I(y_j(x) = i) \tag{3}$$

Here, $f(x)$ is the final prediction of the Bagging classifier for an input x, and $I(\cdot)$ is the indicator function. The Bagging classifier assigns the class that receives the most votes from the base classifiers as the predicted class for input x.

Support vector machine

Support vector machine (SVM) is a traditional ML technique that applies supervised learning models to resolve classification, outlier detection, and regression problems by utilizing data transformations to define boundaries between data points based on predetermined classes, labels, or outputs (*Abdiansah & Wardoyo, 2015*). SVM has a class used for classification known as SVC. The SVM algorithm aims to find a hyperplane that can effectively distinguish between different classes of data points. The hyperplane is positioned in such a way that the distance between the classes is maximized. In a binary classification problem, the SVM algorithm aims to find the hyperplane using Eq. (4).

$$m \cdot x + c = 0 \tag{4}$$

where m is a weight vector, x is a feature vector, c is a bias term and $\cdot$ denotes the dot product between m and x. The hyperplane separates the two classes with maximum margin.

Light gradient boosting machine

Light gradient boosting machine (LGBM) is a machine learning framework developed by Microsoft Research Asia that utilizes a decision-tree based algorithm to enable efficient and distributed training, widely used for classification tasks (*Ke et al., 2017*). This framework is designed to be fast and minimize memory usage while performing gradient boosting, a popular technique in machine learning. LGBM implements two novel strategies, namely gradient-based one side sampling (GOSS) and exclusive feature bundling (EFB), to overcome the limitations in previous gradient boosting decision tree (GBDT) frameworks (*Ahamed & Arya, 2021*). These strategies aim to improve the performance and efficiency of LGBM.

GOSS places a stronger emphasis on the under-trained portion of the dataset, aiming to learn more aggressively from those instances. It achieves this by considering every instance with a higher gradient and applying random sampling on instances with lower gradients.

The variance gain for GOSS is given in Eq. (5). On the other hand, EFB is a distributed and high-performance gradient boosting framework within LGBM. It utilizes a decision tree algorithm with reduced memory usage and the capability to handle large-scale data effectively.

$$\tilde{Var}_y(d) = \frac{1}{m}\left(\frac{\left(\sum_{u_x \in A_p} g_x + \frac{1-a}{b}\sum_{u_x \in B_p} g_x\right)^2}{m_p^y(d)} + \frac{\left(\sum_{u_x \in A_q} g_x + \frac{1-a}{b}\sum_{u_x \in B_q} g_x\right)^2}{m_q^y(d)}\right). \tag{5}$$

XGBoost classifier

XGBoost is an ensemble-based boosting technique that belongs to the distributed machine learning community (DMLC) (*Bhati et al., 2021*). It is known for its high efficiency and meticulous examination of data values in a database. The approach involves constructing a sequential decision tree, also referred to as a sequential ensemble technique.

In this method, each data value in the database is assigned a weight that determines its likelihood of being selected by a decision tree for further analysis. Initially, all data values have the same weight, which is subsequently adjusted based on the analysis. The outcomes of the first iteration contribute to the creation of a new classification model that builds on the previous results. This iterative process continues until the final classifier is formed. Given a fixed tree structure, the optimal weight for each leaf and the resulting objective value are given in Eqs. (6) and (7) respectively (*Guo et al., 2020*).

$$w_I = -\frac{G_I}{H_i + \lambda} \tag{6}$$

$$O = -\frac{1}{2}\sum_{i=1}^{N}\frac{G_i^2}{H_i + \lambda} + \gamma N. \tag{7}$$

Proposed approach: RoBERTa-Large + XGBoost

Our PA leverages a pretrained RoBERTa-Large model for feature extraction, paired with an XGBoost classifier for final classification. The RoBERTa model generates contextualized embeddings for each tweet, capturing nuanced linguistic patterns in hate speech and offensive language. These embeddings are pooled into a single feature vector for each text, which is then used as input to the XGBoost model. The XGBoost classifier further enhances model performance by leveraging RoBERTa's embeddings, optimizing for high classification accuracy. This combination has demonstrated significant improvements over the baseline models in our evaluations.

Model evaluation metrics

Model evaluation is a technique used to analyze the effectiveness of a model based on some constraints (*Sheikh, Goel & Kumar, 2020*). However, while evaluating the model, it should be kept in mind that it cannot overfit or step on the model. To measure the performance of a model, there are several evaluation metrics available such as confusion metrics, accuracy, precision, recall, F1-score *etc.*

In binary classification, accuracy is measured concerning both positive and negative outcomes using the following formula:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{8}$$

where TP = true positives, TN = true negatives, FP = false positives, and FN = false negatives.

The precision value, indicating the accuracy of our model, is given as follows:

$$Precision = \frac{TP}{TP + FP}. \tag{9}$$

Recall (TPR) measures the proportion of actual positive instances that are correctly identified by the model. A high recall indicates that the model is good at identifying positive instances.

$$Recall = \frac{TP}{TP + FN}. \tag{10}$$

F1-score is the harmonic mean (HM) of the precision and recall scores.

$$F1\text{-score} = \frac{2}{\frac{1}{Precision} + \frac{1}{Recall}}$$
$$F1\text{-score} = \frac{2 \times Precision \times Recall}{Precision + Recall}. \tag{11}$$

EXPERIMENT AND RESULTS

In this section, we outline the experimental setup of our study and present a comparative analysis of the results obtained from multiple ML algorithms. The PA of our study is the combination of a pretrained RoBERTa-Large model with an XGBoost classifier, which demonstrated superior performance compared to previous methods. This hybrid approach was benchmarked against six traditional ML algorithms, showcasing its effectiveness in hate speech detection.

Experimental setup

The architecture of the hybrid model consisted of a RoBERTa-Large backbone (355M parameters) modified with an added dropout layer (p = 0.3) and a linear classification head. The transformer component was fine-tuned using AdamW optimization (learning rate $= 2 \times 10^{-5}$, $\beta_1 = 0.9$, $\beta_2 = 0.999$) with a batch size of 18 and a sequence length capped at 130 tokens. Training ran for five epochs with cross-entropy loss, using a linear learning rate scheduler without warm-up steps. The XGBoost classifier was trained on 1,024-dimensional RoBERTa embeddings with the following configuration: learning rate = 0.01, n-estimators = 400, max-depth = 4, and multi-class log loss objective.

All experiments were conducted on dual NVIDIA RTX 3080 GPUs (10 GB VRAM each) with PyTorch 1.10 and XGBoost 1.5, utilizing full FP32 precision. The implementation leveraged HuggingFace Transformers for RoBERTa integration and scikit-learn for evaluation metrics. Hyperparameters such as learning rate, batch size, and

Table 5 Performance comparison of the PA, RoBERTa-Large + XGBoost, with state-of-the-art ML classifiers on the TT dataset in terms of accuracy (Acc.), precision (Prec.), recall (Rec.), F1-score (F1-scr).

Classifier	Acc.	Prec.	Rec.	F1-scr.
RFC	0.93	0.93	0.93	0.93
SVC	0.94	0.94	0.94	0.94
Adaboost	0.91	0.92	0.91	0.91
Bagging	0.94	0.93	0.93	0.93
LGBM	0.93	0.94	0.93	0.93
XGBoost	0.94	0.94	0.94	0.94
PA	0.95	0.95	0.95	0.95

number of epochs were manually tuned based on validation set performance. We found that training for fewer than five epochs resulted in underfitting, while training for more epochs led to overfitting. To mitigate overfitting, a dropout rate of 0.3 was applied, and validation accuracy was closely monitored throughout training.

Results

While we experimented with a total of nine ML models, we specifically focused on discussing and highlighting the performance of the models that achieved superior results. Tables 5, 6 and 7 display the performance of each classifier, including SVC, RFC, BC, AdaBoost, LGBM, and XGBoost trained to detect hate speech and offensive language across the TT, Davidson, and HSD datasets. The results highlight the comparative advantage of our RoBERTa-Large + XGBoost approach in terms of accuracy and contextual understanding.

The results on the TT dataset, shown in Table 5, are presented in terms of accuracy, precision, recall, and F1-score for each classifier. The PA, utilizing RoBERTa-Large embeddings with an XGBoost classifier, outperformed all other models, achieving the highest accuracy of 0.92 and an F1-score of 0.91. This indicates that the PA model provides a more balanced performance in detecting hate speech and offensive language with improved precision and recall as well. Among the other classifiers, both LGBM and traditional XGBoost achieved high accuracy scores of 0.91, with corresponding F1-scores of 0.90. Although SVC showed slightly better recall (0.91) than the other models, its overall performance metrics did not surpass those of the PA model. This suggests that RoBERTa-Large embeddings contribute additional contextual understanding that enhances the model's ability to accurately classify nuanced language in tweets, giving it an edge over traditional models using TF-IDF-based features.

Similarly, the results obtained on the Davidson dataset, presented in Table 6, show that the PA, combining RoBERTa-Large embeddings with XGBoost, achieved the highest performance across all metrics. PA recorded an accuracy of 0.92 and an F1-score of 0.91, outperforming all other models. Both LGBM and traditional XGBoost followed closely with accuracy scores of 0.91, although they fell short in precision and recall. The superior performance of PA is likely due to RoBERTa-Large's ability to capture contextual nuances

Table 6 Performance comparison of the PA, RoBERTa-Large + XGBoost, with state-of-the-art ML classifiers on Davidson dataset in terms of accuracy (Acc.), precision (Prec.), recall (Rec.), F1-score (F1-scr).

Classifier	Acc.	Prec.	Rec.	F1-scr.
RFC	0.90	0.89	0.90	0.88
SVC	0.90	0.90	0.91	0.89
Adaboost	0.90	0.89	0.90	0.89
Bagging	0.90	0.89	0.90	0.90
LGBM	0.91	0.90	0.91	0.90
XGBoost	0.91	0.90	0.90	0.90
PA	0.92	0.91	0.92	0.91

Table 7 Performance comparison of the PA, RoBERTa-Large + XGBoost, with state-of-the-art ML classifiers on HSD dataset in terms of accuracy (Acc.), precision (Prec.), recall (Rec.), F1-score (F1-scr).

Classifier	Acc.	Prec.	Rec.	F1-scr.
RFC	0.95	0.95	0.95	0.95
SVC	0.87	0.88	0.87	0.85
Adaboost	0.94	0.94	0.94	0.94
Bagging	0.94	0.94	0.94	0.94
LGBM	0.89	0.90	0.89	0.87
XGBoost	0.93	0.94	0.93	0.92
PA	0.97	0.96	0.97	0.96

in language, which enhances classification accuracy when paired with XGBoost's robust handling of complex feature relationships. This combination allows for more precise detection of hate speech and offensive language compared to traditional TF-IDF based models.

At last, Table 7 shows that the PA achieved the highest scores on the HSD dataset, with an accuracy of 0.97 and an F1-score of 0.96. RFC followed with strong results across all metrics at 0.95, while Adaboost and Bagging also performed well, each scoring 0.94 across metrics. XGBoost performed slightly lower, achieving an accuracy of 0.93.

RESULT ANALYSIS

Figures 7, 8, and 9 present bar charts illustrating the performance of various models on the TT, Davidson, and HSD datasets, respectively. The evaluated performance metrics include accuracy, precision, recall, and F1-score. The x-axis represents the different models, while the y-axis indicates the corresponding values of each performance metric. In Fig. 7, we observe that for the TT dataset, the PA model achieves the highest accuracy, recall, and F1-score, indicating that it outperforms the other models across these key metrics. The second-best model on the TT dataset is SVC, which slightly outperforms XGBoost in terms of precision.

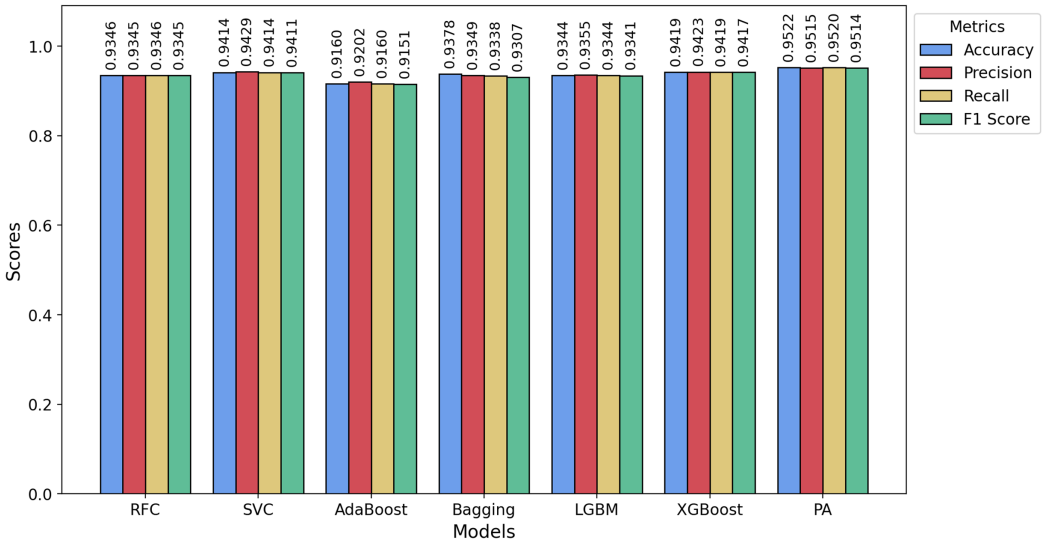

Figure 7 **Bar chart showing the accuracy, precision, recall, and F1-score achieved by the PA and other ML models on the TT dataset.** Full-size DOI: 10.7717/peerj-cs.3214/fig-7

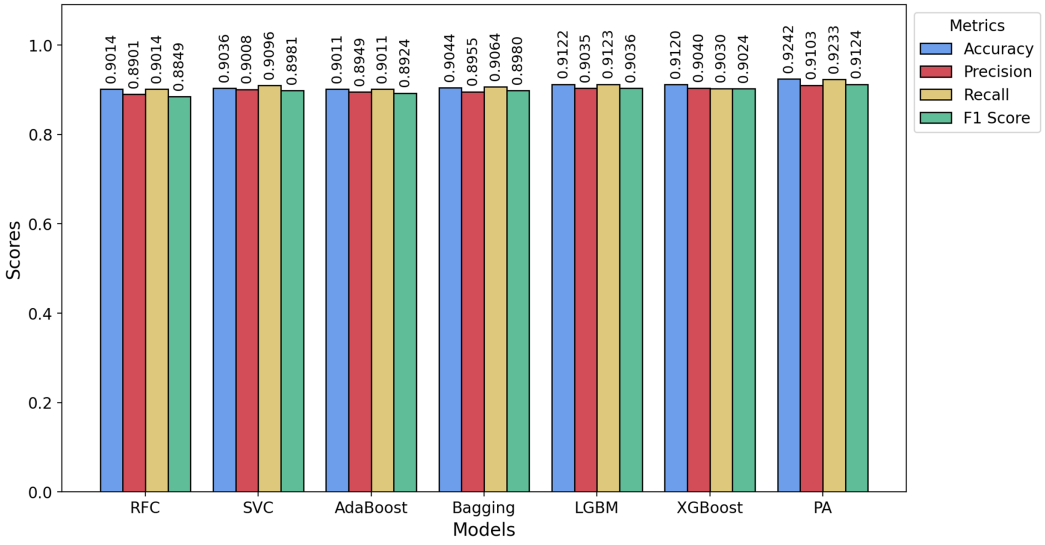

Figure 8 **Bar chart showing the accuracy, precision, recall, and F1-score achieved by the PA and other ML models on the Davidson dataset.** Full-size DOI: 10.7717/peerj-cs.3214/fig-8

For the Davidson dataset, Fig. 8 demonstrates that the PA model consistently achieves the best performance across all metrics, outperforming all other models. Similarly, Fig. 9 highlights that PA surpasses all other classifiers on the HSD dataset as well. Overall, PA demonstrates significant effectiveness in accurately classifying text as hate or non-hate across all three datasets, establishing itself as the top-performing model in this comparative analysis. Figure 10 presents the confusion matrices for the PA model across all datasets. In Fig. 10A, we see that out of approximately 11,349 test samples in the TT dataset, around 553 are misclassified. Figure 10B shows that, for the Davidson dataset, 417 out of 4,957 test

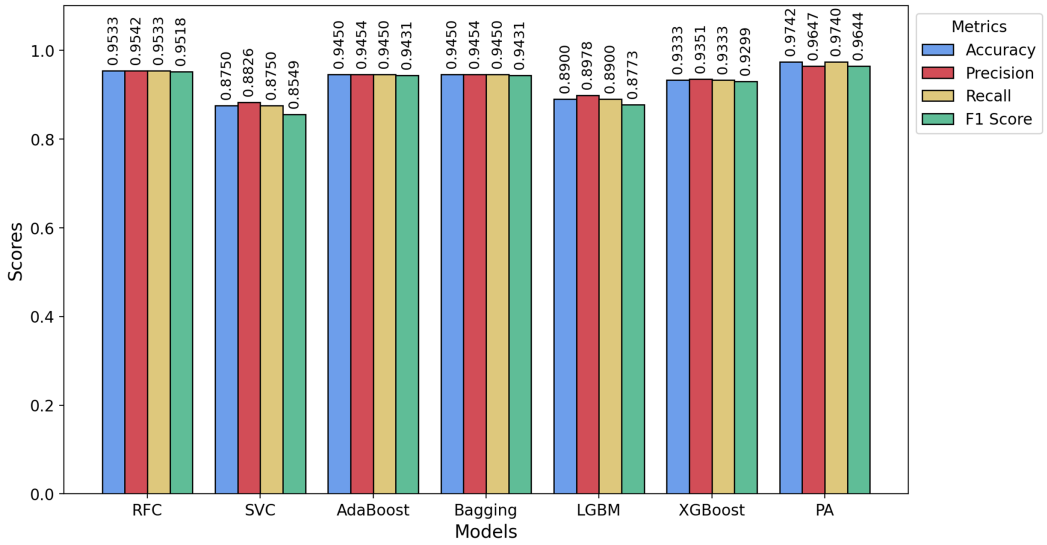

Figure 9 **Bar chart showing the accuracy, precision, recall, and F1-score achieved by the PA and other ML models on the HSD dataset.** Full-size DOI: 10.7717/peerj-cs.3214/fig-9

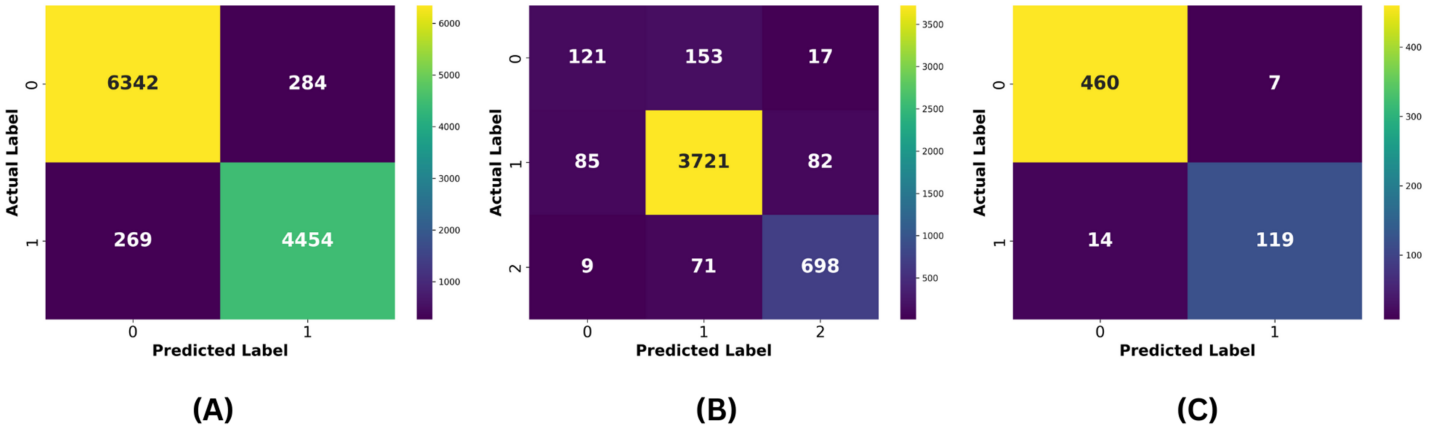

Figure 10 **(A) Confusion matrix of the proposed model for the TT dataset, (B) confusion matrix of the proposed model for the Davidson dataset, and (C) confusion matrix of the proposed model for the HSD dataset.** Full-size DOI: 10.7717/peerj-cs.3214/fig-10

samples are misclassified. In Fig. 10C, we observe that, for the HSD dataset, only 21 out of 600 test samples are misclassified. These confusion matrices indicate that the PA model performs exceptionally well on the TT and HSD datasets, with minimal misclassification.

However, the Davidson dataset has a relatively higher rate of misclassification compared to the other datasets. This discrepancy can likely be attributed to two factors: (1) the multiclass nature of the Davidson dataset, which increases classification complexity, and (2) the imbalanced class distribution within the dataset, which skews the model's predictions. Despite these challenges, the PA model still demonstrates strong performance on the Davidson dataset, showing its robustness. The remaining misclassifications across all datasets may be due to the uneven distribution of samples within each dataset, which can impact the model's generalization ability.

Table 8 Comparison of the results obtained by the state-of-the-art approaches with PA on Davidson dataset in terms of accuracy (Acc.), precision (Prec.), recall (Rec.), F1-score (F1-scr).

Approach	Acc.	Prec.	Rec.	F1-scr.
DvdW	0.89	0.91	0.90	0.90
DvdB	–	Hateful = 0.32, Offensive = 0.96, Neither = 0.81	Hateful = 0.53, Offensive = 0.88, Neither = 0.95	Hateful = 0.40, Offensive = 0.92, Neither = 0.81
Mrk	–	–	–	Hateful = 0.53, Offensive = 0.85, Neither = 0.92
Mtg	0.92	0.75	0.75	0.75
Mosc	–	–	–	0.886
Awl	–	0.9071	0.9114	0.9071
Ali (RoBERTa)	0.78	–	–	0.74
Mlk (ALBERT)	–	0.90	0.91	0.90
Mlk (ELECTRA)	–	0.91	0.91	0.91
PA	0.9242	0.9103	0.9233	0.9124

Table 8 shows the comparison of the PA in terms of accuracy, precision, recall, and F1-score with different state-of-the-art approaches, namely: DvdW (*Davidson et al., 2017*), DvdB (*Davidson, Bhattacharya & Weber, 2019*), Mrk (*Maronikolakis, Baader & Schütze, 2022*), Mtg (*Mutanga, Naicker & Olugbara, 2020*), Mosc (*Mosca, Wich & Groh, 2021*), Awl (*Awal et al., 2021*), Ali (*Ali, Blackburn & Stringhini, 2025*) and Mlk (*Malik et al., 2024*). The comparison is only for the Davidson dataset because, to the best of our knowledge, other two datasets are relatively new ones and there is no promising work done on these two datasets. TT dataset is a combination of various datasets that have a high level of class imbalance in them, and the HSD dataset is the latest dataset in the field of hate speech detection, so the approach of this article is the baseline for other researchers to work on these datasets.

Table 8 shows that our approach exhibited significantly superior performance compared to the state-of-the-art approaches. The state-of-the-art approaches listed in Table 8 encompassed a variety of ML, DL, and Transformer-based models. DvdW (*Davidson et al., 2017*), DvdB (*Davidson, Bhattacharya & Weber, 2019*), and Mosc (*Mosca, Wich & Groh, 2021*) employed ML and NLP techniques, while Mrk (*Maronikolakis, Baader & Schütze, 2022*) and Awl (*Awal et al., 2021*) implemented DL models, and Mtg (*Mutanga, Naicker & Olugbara, 2020*), Ali (*Ali, Blackburn & Stringhini, 2025*) and Mlk *Malik et al. (2024)* leveraged Transformer-based model for the detection of hate speech and offensive language for the Davidson dataset. Nonetheless, the PA of this article outperformed all these methods in terms of performance. The PA achieved an accuracy of 92.42%, surpassing the accuracy of 89% achieved by DvdW and 92% by Mtg. The PA also surpassed other approaches in terms of precision, recall, and F1-score. The improved performance of our approach can be attributed to two main factors. Firstly, we conducted thorough preprocessing on the dataset to ensure its cleanliness and quality before feeding it

into the model for training. This preprocessing step played a significant role in enhancing the results.

Secondly, we utilized a hybrid approach, combining the RoBERTa-large model with XGBoost for classification. This approach leverages RoBERTa's ability to capture deep contextual language features and the robustness of XGBoost, especially effective for datasets with imbalanced class distributions like the Davidson dataset. Fine-tuning of the hyperparameters further optimized the model performance, contributing to these improved results. These findings suggest that our approach is a promising tool for detecting hate speech and offensive language in online forums.

CONCLUSION AND DISCUSSION

This article focuses on hate speech detection in social media. While hate speech as a social issue is an established research topic in the arts and humanities, it is still a relatively new topic in the computing sector. Hate speech is a very serious issue as it has many drastic impacts on society. Therefore, it is necessary to update researchers frequently on new developments or advancements in this area. We explored the techniques employed by conventional ML methods for identifying hate speech on social media platforms.

Before carrying out our research, we identified which social media platform is the most toxic, so we chose that one for our study. We found that Twitter is the most toxic social media platform, and English is the most popular language on Twitter. This study involves the detection of hate speech and offensive language detection in three different datasets of the English language consisting of tweets from different social media platforms. Two of the three datasets used in this study are novel contributions to the field, as no previous research has been conducted on these specific datasets. Our work is a baseline for the other researchers to work on these two datasets.

In this study, we compared our results with state-of-the-art approaches in hate speech detection, including a variety of ML, DL, and Transformer-based techniques. By leveraging the RoBERTa-large model to capture contextual features and replacing its classification layer with a hyperparameter-tuned XGBoost model, we achieved 92.42% accuracy on the Davidson dataset, surpassing prior work. This hybrid approach demonstrates that combining Transformer-based contextual understanding with ensemble learning (XGBoost) can outperform pure Transformer or traditional ML methods. The RoBERTa-large model excels at capturing nuanced linguistic patterns (*e.g.*, sarcasm, implicit hate), while XGBoost refines these features through its gradient-boosting framework, optimizing decision boundaries for class imbalance, a common challenge in hate speech datasets. This synergy offers a new direction for hate speech detection systems, where Transformers handle semantic depth and ensembles improve robustness to data variability.

Our work has practical implications, such as the proposed model in our work can be deployed to automate the moderation activity of social media platforms more accurately, reducing exposure to harmful content while minimizing false positives. By improving detection accuracy, our approach could help create safer online spaces, particularly for marginalized communities that are disproportionately targeted by hate speech.

Furthermore, reducing the reliance on human moderators for initial screening could minimize their exposure to psychologically harmful content while maintaining the nuanced judgment needed for borderline cases. Future implementations could also provide transparency reports to help platforms demonstrate their commitment to combating online hate.

Limitations and future work

This study has certain limitations that should be acknowledged:

Language-specific focus: The scope of this research is restricted to English-language tweets, which limits its applicability to other languages or multilingual social media platforms.

Data bias and ethics: The datasets used in this study, including two relatively new datasets (TT and HSD), may not fully capture the diversity of hateful content across different platforms, regions, or time periods. As pre-annotated datasets, they are also subject to annotation subjectivity and cultural biases, which can affect model fairness and reliability. The novelty of TT and HSD means that their annotation quality and bias profiles are less well-established than those of benchmark datasets like Davidson. Finally, deploying automated hate speech detection systems carries risks of misclassification, which require careful consideration in real-world use.

Modality limitation: The model is designed to detect hate speech only from textual data, which excludes multimodal forms of hate speech commonly shared on social media, such as hateful images, videos, or memes.

Interpretability consideration: While the PA achieves strong predictive performance, the current study does not incorporate *post-hoc* explainability tools such as SHapley Additive exPlanations (SHAP) and Local Interpretable Model-Agnostic Explanations (LIME) or attention visualizations. Including such tools in future work could further enhance the transparency and trustworthiness of the model's decisions, particularly in sensitive application areas.

Building on identified limitations, future research could broaden the scope by incorporating multimodal data, such as images and videos alongside text, to enable a more comprehensive analysis of hate speech on social media platforms. Additionally, future studies could explore hate speech detection in multiple languages beyond English, such as Hindi, Urdu, and Japanese, to broaden the applicability of these techniques. An important future direction is the inclusion of emojis, which often convey emotional or contextual cues in online communication. We plan to develop an emoji-to-text conversion module and investigate how emoji semantics interact with hate speech content, potentially improving detection performance. We also plan to focus on models that can more effectively address class imbalance issues and perform well on multiclass data, as these are key challenges in hate speech detection.

Further, to enhance transparency and accountability, we aim to integrate interpretability techniques into our future work. This includes applying SHAP and LIME for *post-hoc* inspection, as well as attention visualization mechanisms inherent in transformer architectures. These tools will help elucidate which features or tokens influence classification decisions and ensure the model's behavior aligns with human-understandable reasoning. Given the strong performance of hybrid models, we also aim to leverage this approach for other classification problems, combining model strengths to yield better results than individual models alone. This study is intended as a guide for newcomers to the field, laying out the complete process of text classification using ML and Transformer-based approaches in hate speech detection.

ACKNOWLEDGEMENTS

We express gratitude to the High-Performance Computing (HPC) Lab at the University of Management and Technology (UMT) for providing the computational resources and support necessary to conduct the experiments in this research.

ADDITIONAL INFORMATION AND DECLARATIONS

Funding

This research work is supported by the Faculty of Information Science & Technology, The National University of Malaysia. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Grant Disclosures

The following grant information was disclosed by the authors:
Faculty of Information Science & Technology.
The National University of Malaysia.

Competing Interests

The authors declare that they have no competing interests.

Author Contributions

- Uzair Iftikhar conceived and designed the experiments, performed the experiments, analyzed the data, prepared figures and/or tables, and approved the final draft.
- Syed Farooq Ali conceived and designed the experiments, performed the experiments, analyzed the data, prepared figures and/or tables, and approved the final draft.
- Ghulam Mustafa conceived and designed the experiments, performed the experiments, analyzed the data, performed the computation work, prepared figures and/or tables, and approved the final draft.
- Nurhidayah Bahar performed the computation work, authored or reviewed drafts of the article, and approved the final draft.
- Kashif Ishaq performed the computation work, authored or reviewed drafts of the article, and approved the final draft.

Data Availability

The following information was supplied regarding data availability:

The Hybrid Transformer-Ensemble Approach for Detecting Hate Speech data is available at figshare: Iftikhar, Uzair (2025). Hybrid Transformer-Ensemble Approach for Detecting Hate Speech. figshare. Software. https://doi.org/10.6084/m9.figshare.29757155.v1.

The Davidson Dataset is available at GitHub: https://github.com/t-davidson/hate-speech-and-offensive-language.

The HateSpeechDataset (HSD) is available at figshare: Cooke, Shane (2022). Labelled Hate Speech Detection Dataset. figshare. Dataset. https://doi.org/10.6084/m9.figshare.19686954.v1.

The ToxicTweetDataset (TT) is available at Kaggle: https://www.kaggle.com/ashwiniyer176/toxic-tweets-dataset.

REFERENCES

Abdiansah A, Wardoyo R. 2015. Time complexity analysis of support vector machines (SVM) in LibSVM. *International Journal Computer and Application* **128(3)**:28–34 DOI 10.5120/ijca2015906480.

Abro S, Shaikh S, Khand ZH, Zafar A, Khan S, Mujtaba G. 2020. Automatic hate speech detection using machine learning: a comparative study. *International Journal of Advanced Computer Science and Applications* **11(8)**:61 DOI 10.14569/IJACSA.2020.0110861.

Ahamed BS, Arya S. 2021. LGBM classifier based technique for predicting type-2 diabetes. *European Journal of Molecular & Clinical Medicine* **8(3)**:454–467.

Alfina I, Mulia R, Fanany MI, Ekanata Y. 2017. Hate speech detection in the Indonesian language: a dataset and preliminary study. In: *2017 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*. Piscataway: IEEE, 233–238.

Ali S, Blackburn J, Stringhini G. 2025. Evolving hate speech online: an adaptive framework for detection and mitigation. ArXiv DOI 10.48550/arXiv.2502.10921.

Aljero MKA, Dimililer N. 2021. A novel stacked ensemble for hate speech recognition. *Applied Sciences* **11(24)**:11684 DOI 10.3390/app112411684.

Ansari G, Kaur P, Saxena C. 2024. Data augmentation for improving explainability of hate speech detection. *Arabian Journal for Science and Engineering* **49(3)**:3609–3621 DOI 10.1007/s13369-023-08100-4.

Anti-Defamation League. 2025. Online hate and harassment: the American experience 2021. *Available at https://www.adl.org/resources/report/online-hate-and-harassment-american-experience-2021* (accessed 5 March 2023).

Awal MR, Cao R, Lee RK-W, Mitrović S. 2021. AngryBERT: joint learning target and emotion for hate speech detection. In: *Advances in Knowledge Discovery and Data Mining: 25th Pacific-Asia Conference, PAKDD 2021, Virtual Event, May 11–14, 2021, Proceedings, Part I*. Cham: Springer, 701–713.

Bhati BS, Chugh G, Al-Turjman F, Bhati NS. 2021. An improved ensemble based intrusion detection technique using XGBoost. *Transactions on Emerging Telecommunications Technologies* **32(6)**:e4076 DOI 10.1002/ett.4076.

Biradar S, Saumya S, Chauhan A. 2022. Fighting hate speech from bilingual hinglish speaker's perspective, a transformer-and translation-based approach. *Social Network Analysis and Mining* **12(1)**:87 DOI 10.1007/s13278-022-00920-w.

Breiman L. 1996. Bagging predictors. *Machine Learning* **24(2)**:123–140 DOI 10.1023/a:1018054314350.

Breiman L. 2001. Random forests. *Machine Learning* **45(1)**:5–32 DOI 10.1023/A:1010933404324.

Brown TB, Mann B, Ryder N, Subbiah M, Kaplan J, Dhariwal P, Neelakantan A, Shyam P, Sastry G, Askell A, Agarwal S, Herbert-Voss A, Krueger G, Henighan T, Child R, Ramesh A, Ziegler DM, Wu J, Winter C, Hesse C, Chen M, Sigler E, Litwin M, Gray S, Chess B, Clark J, Berner C, McCandlish S, Radford A, Sutskever I, Amodei D. 2020. Language models are few-shot learners. *Advances in Neural Information Processing Systems* **33**:1877–1901.

Cooke S. 2022. Labelled hate speech detection dataset. *Available at https://figshare.com*.

Davidson T, Bhattacharya D, Weber I. 2019. Racial bias in hate speech and abusive language detection datasets. ArXiv DOI 10.48550/arXiv.1905.12516.

Davidson T, Warmsley D, Macy M, Weber I. 2017. Automated hate speech detection and the problem of offensive language. *Proceedings of the International AAAI Conference on Web and Social Media* **11(1)**:512–515 DOI 10.1609/icwsm.v11i1.14955.

del Valle-Cano G, Quijano-Sánchez L, Liberatore F, Gómez J. 2023. SocialHaterBERT: a dichotomous approach for automatically detecting hate speech on twitter through textual analysis and user profiles. *Expert Systems with Applications* **216(12)**:119446 DOI 10.1016/j.eswa.2022.119446.

Dinakar K, Reichart R, Lieberman H. 2011. Modeling the detection of textual cyberbullying. *Proceedings of the International AAAI Conference on Web and Social Media* **5(3)**:11–17 DOI 10.1609/icwsm.v5i3.14209.

Fortuna P, Nunes S. 2018. A survey on automatic detection of hate speech in text. *ACM Computing Surveys (CSUR)* **51(4)**:1–30 DOI 10.1145/3232676.

Gambäck B, Sikdar UK. 2017. Using convolutional neural networks to classify hate-speech. In: *Proceedings of the First Workshop on Abusive Language Online*, 85–90.

Ghosh S, Ghosh S, Das D. 2017. Sentiment identification in code-mixed social media text. ArXiv DOI 10.48550/arXiv.1707.01184.

Githa IPWN, Syananda A, Faustine R, Edbert IS, Suhartono D. 2024. Hate speech classification in Indonesian tweets using TF-IDF and data augmentation. In: *2024 International Conference on Green Energy, Computing and Sustainable Technology (GECOST)*. Piscataway: IEEE, 61–65.

Greevy E, Smeaton AF. 2004. Classifying racist texts using a support vector machine. In: *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. New York: ACM, 468–469.

Guo J, Dai Y, Wang C, Wu H, Xu T, Lin K. 2020. A physiological data-driven model for learners' cognitive load detection using HRV-PRV feature fusion and optimized XGBoost classification. *Software: Practice and Experience* **50(11)**:2046–2064 DOI 10.1002/spe.2730.

Ishmam AM, Sharmin S. 2019. Hateful speech detection in public Facebook pages for the Bengali language. In: *2019 18th IEEE International Conference on Machine Learning and Applications (ICMLA)*. Piscataway: IEEE, 555–560.

Iyer AU. 2021. Toxic tweets dataset. *Available at https://www.kaggle.com*.

Ke G, Meng Q, Finley T, Wang T, Chen W, Ma W, Ye Q, Liu T-Y. 2017. LightGBM: a highly efficient gradient boosting decision tree. *Advances in Neural Information Processing Systems* **30**:3149–3157.

Khanday AMUD, Rabani ST, Khan QR, Malik SH. 2022. Detecting twitter hate speech in COVID-19 era using machine learning and ensemble learning techniques. *International Journal of Information Management Data Insights* **2(2)**:100120 DOI 10.1016/j.jjimei.2022.100120.

Khullar A, Nkemelu D, Nguyen VC, Best ML. 2024. Hate speech detection in limited data contexts using synthetic data generation. *ACM Journal on Computing and Sustainable Societies* **2(1)**:1–18 DOI 10.1145/3625679.

Köffer S, Riehle DM, Höhenberger S, Becker J. 2018. Discussing the value of automatic hate speech detection in online debates. In: *Multikonferenz Wirtschaftsinformatik (MKWI 2018): Data Driven X-Turning Data in Value, Leuphana, Germany*.

Kwok I, Wang Y. 2013. Locate the hate: detecting tweets against blacks. *Proceedings of the AAAI Conference on Artificial Intelligence* **27(1)**:1621–1622 DOI 10.1609/aaai.v27i1.8539.

Liu S, Forss T. 2014. Combining n-gram based similarity analysis with sentiment analysis in web content classification. In: *Special Session on Text Mining*. Vol. 2. Setúbal: SciTePress, 530–537.

Madaan M, Kumar A, Keshri C, Jain R, Nagrath P. 2021. Loan default prediction using decision trees and random forest: a comparative study. *IOP Conference Series: Materials Science and Engineering* **1022**:12042 DOI 10.1088/1757-899x/1022/1/012042.

Mahesh B. 2020. Machine learning algorithms—a review. *International Journal of Science and Research (IJSR)* **9(1)**:381–386 DOI 10.21275/ART20203995.

Malik JS, Qiao H, Pang G, van den Hengel A. 2024. Deep learning for hate speech detection: a comparative study. *International Journal of Data Science and Analytics* **39(1)**:1–16 DOI 10.1007/s41060-024-00650-6.

Malmasi S, Zampieri M. 2017. Detecting hate speech in social media. ArXiv DOI 10.48550/arXiv.1712.06427.

Mansur Z, Omar N, Tiun S. 2023. Twitter hate speech detection: a systematic review of methods, taxonomy analysis, challenges, and opportunities. *IEEE Access* **11**:16226–16249 DOI 10.1109/access.2023.3239375.

Mansur Z, Omar N, Tiun S, Alshari EM. 2024. A normalization model for repeated letters in social media hate speech text based on rules and spelling correction. *PLOS ONE* **19(3)**:e0299652 DOI 10.1371/journal.pone.0299652.

Markov I, Gevers I, Daelemans W. 2022. An ensemble approach for dutch cross-domain hate speech detection. In: *Natural Language Processing and Information Systems: 27th International Conference on Applications of Natural Language to Information Systems, NLDB 2022, Valencia, Spain, June 15–17, 2022, Proceedings*. Cham: Springer, 3–15.

Maronikolakis A, Baader P, Schütze H. 2022. Analyzing hate speech data along racial, gender and intersectional axes. ArXiv DOI 10.48550/arXiv.2205.06621.

Mathew B, Saha P, Yimam SM, Biemann C, Goyal P, Mukherjee A. 2021. HateXplain: a benchmark dataset for explainable hate speech detection. *Proceedings of the AAAI Conference on Artificial Intelligence* **35(17)**:14867–14875 DOI 10.1609/aaai.v35i17.17745.

Mosca E, Wich M, Groh G. 2021. Understanding and interpreting the impact of user context in hate speech detection. In: *Proceedings of the Ninth International Workshop on Natural Language Processing for Social Media*, 91–102.

Mozafari M, Farahbakhsh R, Crespi N. 2020. A BERT-based transfer learning approach for hate speech detection in online social media. In: *Complex Networks and Their Applications VIII: Volume 1 Proceedings of the Eighth International Conference on Complex Networks and Their Applications COMPLEX NETWORKS 2019 8*. Cham: Springer, 928–940.

Muhammad A, Abdullah S, Sani NS. 2021. Optimization of sentiment analysis using teaching-learning based algorithm. *Computers, Materials & Continua* **69(2)**:1783–1799 DOI 10.32604/cmc.2021.018593.

Mullah NS, Zainon WMNW. 2021. Advances in machine learning algorithms for hate speech detection in social media: a review. *IEEE Access* **9**:88364–88376 DOI 10.1109/access.2021.3089515.

Mutanga RT, Naicker N, Olugbara OO. 2020. Hate speech detection in twitter using transformer methods. *International Journal of Advanced Computer Science and Applications* **11(9)**:614–620 DOI 10.14569/IJACSA.2020.0110972.

Nobata C, Tetreault J, Thomas A, Mehdad Y, Chang Y. 2016. Abusive language detection in online user content. In: *Proceedings of the 25th International Conference on World Wide Web*, 145–153.

Norton L. 2022. America ranks the most toxic social media apps. *Available at https://simpletexting.com/blog/most-toxic-social-media-apps/* (accessed 5 March 2023).

Othman H, Yaakub MR. 2022. Implementing BERT with K-Train library for sentiment analysis. *Journal of Visual Languages & Computing* **2022(2)**:26–34 DOI 10.18293/JVLC2022-N2-024.

Pereira-Kohatsu JC, Quijano-Sánchez L, Liberatore F, Camacho-Collados M. 2019. Detecting and monitoring hate speech in twitter. *Sensors* **19(21)**:4654 DOI 10.3390/s19214654.

Philipo AG, Ding J, Sarwatt DS, Mohamed JA, Yusufu AS, Daneshmand M, Ning H. 2025. Sentiment-based methods for cyberbullying detection. *Authorea Preprints*.

Pitsilis GK, Ramampiaro H, Langseth H. 2018. Effective hate-speech detection in twitter data using recurrent neural networks. *Applied Intelligence* **48(12)**:4730–4742 DOI 10.1007/s10489-018-1242-y.

Raffel C, Shazeer N, Roberts A, Lee K, Narang S, Matena M, Zhou Y, Li W, Liu PJ. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research* **21(140)**:1–67.

Ramesh A, Dhariwal P, Nichol A, Chu C, Chen M. 2022. Hierarchical text-conditional image generation with clip latents. ArXiv DOI 10.48550/arXiv.2204.06125.

Rawat V, Suryakant S. 2019. A classification system for diabetic patients with machine learning techniques. *International Journal of Mathematical, Engineering and Management Sciences* **4(3)**:729–736 DOI 10.33889/IJMEMS.2019.4.3-057.

Ruby D. 2023. 58+ twitter statistics for marketers in 2023 (users & trends). *Available at https://www.demandsage.com* (accessed 5 March 2023).

Saharia C, Chan W, Saxena S, Li L, Whang J, Denton E, Seyed Ghasemipour SK, Karagol Ayan B, Mahdavi SS, Gontijo-Lopes R, Salimans T, Ho J, Fleet DJ, Norouzi M. 2022. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems* **35**:36479–36494 DOI 10.1145/3528233.3530757.

Sharma S, Agrawal S, Shrivastava M. 2018. Degree based classification of harmful speech using twitter data. ArXiv DOI 10.48550/arXiv.1806.04197.

Sheikh MA, Goel AK, Kumar T. 2020. An approach for prediction of loan approval using machine learning algorithm. In: *2020 International Conference on Electronics and Sustainable Communication Systems (ICESC)*. Piscataway: IEEE, 490–494.

Shen J, Pang R, Weiss RJ, Schuster M, Jaitly N, Yang Z, Chen Z, Zhang Y, Wang Y, Skerry-Ryan RJ, Saurous RA, Agiomyrgiannakis Y, Wu Y. 2018. Natural TTS synthesis by conditioning WaveNet on MEL spectrogram predictions. In: *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Piscataway: IEEE, 4779–4783.

Shepard J. 2023. 23 essential twitter statistics you need to know in 2023. *Available at https://thesocialshepherd.com/blog/twitter-statistics* (accessed 20 May 2023).

Sigurbergsson GI, Derczynski L. 2019. Offensive language and hate speech detection for Danish. ArXiv DOI 10.48550/arXiv.1908.04531.

Singh S. 2025. X (Twitter) statistics 2025: active users & demographics. *Available at https://www.demandsage.com/twitter-statistics/#:~:text=Twitter%20has%20around%20450%20million%20monthly%20active%20users%20as%20of%202023%E2%80%9D*.

Sukawai E, Omar N. 2020. Corpus development for Malay sentiment analysis using semi supervised approach. *Asia–Pacific Journal of Information Technology and Multimedia* **9(1)**:94–109 DOI 10.17576/apjitm-2020-0901-08.

Velankar A, Patil H, Joshi R. 2022. Mono vs multilingual BERT for hate speech detection and text classification: a case study in Marathi. In: *Artificial Neural Networks in Pattern Recognition: 10th IAPR TC3 Workshop, ANNPR 2022, Dubai, United Arab Emirates, November 24–26, 2022, Proceedings*. Cham: Springer, 121–128.

Vidgen B, Yasseri T. 2020. Detecting weak and strong islamophobic hate speech on social media. *Journal of Information Technology & Politics* **17(1)**:66–78 DOI 10.1080/19331681.2019.1702607.

Waseem Z, Hovy D. 2016. Hateful symbols or hateful people? Predictive features for hate speech detection on twitter. In: *Proceedings of the NAACL Student Research Workshop*, 88–93.

Washburn PS, Mahendran, Dhanasekharan, Periyasamy, Murugeswari. 2020. Investigation of severity level of diabetic retinopathy using AdaBoost classifier algorithm. *Materials Today: Proceedings* **33**:3037–3042 DOI 10.1016/j.matpr.2020.03.199.