DBFN-J: A Lightweight and Efficient Model for Hate Speech Detection on Social Media Platforms

Nourah Fahad Janbi*[1], Abdulwahab Ali Almazroi[2], Nasir Ayub[3]
College of Computing and Information Technology at Khulais, Department of Information Technology,
University of Jeddah, Jeddah, 21959, Saudi Arabia[1,2]
Department of Creative Technologies, Air University Islamabad, Islamabad, 44000, Pakistan[3]

***Abstract*—Hate speech on social media platforms like YouTube, Facebook, and Twitter threatens online safety and societal harmony. Addressing this global challenge requires innovative and efficient solutions. We propose DBFN-J (DistillBERT-Feedforward Neural Network with Jaya optimization), a lightweight and effective algorithm for detecting hate speech. This method combines DistillBERT, a distilled version of the Bidirectional Encoder Representations from Transformers (BERT), with a Feedforward Neural Network. The Jaya algorithm is employed for parameter optimization, while aspect-based sentiment analysis further enhances model performance and computational efficiency. DBFN-J demonstrates significant improvements over existing methods such as CNN BERT (Convolutional Neural Network BERT), BERT-LSTM (Long Short-Term Memory), and ELMo (Embeddings from Language Models). Extensive experiments reveal exceptional results, including an AUC (Area Under the Curve) of 0.99, a log loss of 0.06, and a balanced F1-score of 0.95. These metrics underscore its robust ability to identify abusive content effectively and efficiently. Statistical analysis further confirms its precision (0.98) and recall, making it a reliable tool for detecting hate speech across diverse social media platforms. By outperforming traditional algorithms in both performance and resource utilization, DBFN-J establishes a new benchmark for hate speech detection. Its lightweight design ensures suitability for large-scale, resource-constrained applications. This research provides a robust framework for protecting online environments, fostering healthier digital spaces, and mitigating the societal harm caused by hate speech.**

Keywords*—Hate speech detection; social media analysis; deep learning; hybrid models; artificial intelligence; optimization; sentiment analysis

I. Introduction

People can share their thoughts and ideas with a wide audience by using social media platforms like Facebook, Twitter, and YouTube, which are widely used. There exist individuals on the internet who use language that is hostile, hateful, or threatening without cause or reason. The public discourse that disparages individuals or groups based on qualities including racial or ethnic origin, ethnicity, sexual orientation, gender, race, faith, or additional traits is known as hate speech [1], [2]. This presents a serious and persistent problem. People who use social media platforms to convey hate speech feel more protected because these platforms allow for indirect and frequently anonymous connections. Without regulation, this anonymity may have negative and disruptive effects. Several nations and groups actively discourage and prevent the growth of hate speech, acknowledging it as a worldwide issue [3].

Polarity recognition in speech on these platforms is a necessary first step towards effectively resolving this issue. Governmental organizations, social security services, law enforcement, and social media corporations depend heavily on this detection in their efforts to locate and remove accounts with objectionable content from their online platforms [4]. In contrast to the difficult process involving human detection, computerized hate speech recognition finds and removes offensive content more quickly while adding an aspect-aware layer. Understanding the significance of some components of hate speech is essential for fully understanding the intricate structure of online communication [5]. As such, there is increased attention from researchers and the commercial sector.

Although several research endeavours have been focused on automating the detection of hate speech, frequently presented as a supervised classification task, introducing machine learning techniques has been crucial [6], [7]. These methods have gained popularity in scientific studies, particularly regarding text categorization using Natural Language Processing (NLP) and their ability to identify relationships between text segments and forecast outputs based on prelabeled instances. Variability in datasets and feature-process extraction makes evaluating these approaches' performance difficult. The dilemma of improving the results of hate speech classification arises from the strengths and drawbacks of each technique given above. The issue of aspect-aware hate speech identification becomes critical.

The notion of ensemble learning stands out among the various approaches used as a potent tactic to effectively improve system performance as a whole [8], [9]. Ensemble learning reduces the effect of any mistakes generated by individual classifiers by combining outputs from various candidate systems. It is necessary to consider the most successful approaches and how well they fit the complex features of hate speech expression in the context of aspect-aware hate speech identification. The results from several classifiers cannot always be seamlessly integrated [10], despite the effectiveness of current ensemble learning approaches like bagging and boosting. Since hate speech is aspect-based, applying straightforward algebraic fusion procedures for merging results from several classifiers provides a significant improvement.

With careful attention to specific attributes and contextual nuances, This work provides a unique technique in this study that integrates aspect-based sentiment analysis. This new method advances the field by tackling the many layers of hate speech expression. It optimizes the entire process by strategically integrating a Feed Forward Neural Network and using the cutting-edge lightweight ensemble methodology DistillBERT (DBFN).

with the novel ensemble, this model performs rigorous simulations. The technical contributions of this article are:

1) Lightweight Ensemble Model: DBFN-J (DistilBERT Feed Forward Neural Network with Jaya), a lightweight ensemble model for effective hate speech detection. Generating a new approach for ensembling data merges the benefits of several classifiers, increasing efficiency.
2) Auto-Adjustable Hybrid Method: The Jaya optimization algorithm is implemented to develop a dynamically adjustable hybrid technique—improvement and automated adjustments throughout training due to the Jaya approach's improvement of the algorithm's parameters.
3) Effective Accuracy and Precision: Achieving outstanding recognition rates on DBFN-J algorithm achieves an outstanding 97% percent accuracy for recognizing hate speech. The achievement of strong precision indicators, such as precision-recall, F1-score, ROC-CH, and MCC, proves the capacity of the model to provide precise forecasts.
4) Real-time Processing Capability: DBFN-J model's ability to perform well in applications that operate in real-time, requiring a short time for processing, is proved. Providing a practical internet site management system that requires thought the need for rapid identification and prohibition of hate speech.
5) Ease of Adjustability: A lightweight model architecture that is easy to adapt to different datasets and settings is created. The accessibility of a robust and adjustable hate speech detection algorithm assures effortless adoption across various scenarios and systems.
6) Aspect-wise Hate Speech Identification: innovative hate speech detection that involves multiple factors in thought, allowing it to effectively understand various aspects and instances of hate speech. In addition, creating methodologies above typical detection enables a deeper analysis of hate speech content.

Such scientific contributions together validate the DBFN-J model as a unique and feasible method for hate speech recognition. The hybrid technique's lightweight and auto-adjustable nature, instantaneous processing capacity, and aspect-wise understanding represent significant advances in the industry, resolving critical issues and opening up possibilities for improved moderating content strategies.

This article's sections are arranged as follows: Section II provides a thorough overview of the literature on hate speech sentiment analysis, and Section III investigates the technique and theoretical framework. Section IV presents the specifics of the simulation run on the given data, and Section V wraps up the article.

II. Related Work

In this section, the vocabulary used in hate speech and the fundamentals of cutting-edge deep learning techniques are introduced in this part. Furthermore, Table I summarises related work.

A. Hate Speech Terminology

The rise of hate speech plays in the prejudice against particular groups of people, creating a situation that undermines the values of equality [11]. Such targeted Bias mainly affects women and immigrants. Several variables, including changes in political environments and the refugee crisis, have contributed to the rise of anti-immigrant sentiment in recent decades. Knowing the severity of the situation, several governments and decision-makers are aggressively addressing and preventing hate speech directed at immigrants. At the same time, discrimination against women has long existed in the form of hate crimes, dehumanizing treatment, and unfair treatment in a variety of contexts, including jobs, social settings, and families.

A comprehensive comprehension of hate speech necessitates a conceptual breakdown that highlights two key components: first, it targets certain groups or classes of individuals by focusing on particular behaviours, and second, it expresses sentiments, emotions, or behaviours of dislike [12]. Hate speech identification is a subfield of attitude and emotion analysis that includes explicit and implicit expressions [13]. Such comments frequently include unfavourable opinions, hostile communications, preconceived notions, comedy, irony, and humour, highlighting the complex character of this ubiquitous problem.

B. Existing Methods in DL and ML

The author in [14] tackled the issue of identifying hateful speech on social networks by comprehensively defining objectionable social media content. Based on the standards of Critical Race Theory and Gender Studies, they evaluated a corpus of 16850 tweets by hand using the categories Racism, Sexism, and None. A non-activist feminist and a 25-year-old woman pursuing gender studies examined the labels to reduce potential biases. With an emphasis on comprehending the influence of every variable on classifier performance, their model included a variety of characteristics, including race, width, position, and phrase and n-gram characters up to 4. Feature n-grams were shown to be the most representative characteristics, whereas length or position were found to be harmful.

Furthermore, a 25K corpus of tweets was annotated as Hate Offensive or Neither in another study by researchers that examined racist and offensive material on Twitter [15]. Various multiclass classifiers, such as logistic Regression, Random Forests, Naïve Bayes (NB) and Decision Trees, were tested. Term frequency using the Inverse Document Frequency (TF-IDF), balanced n-grams, emotion scores for Part of Speech (POS) identification, and tweet-level material like hashtags, pointing out, responses, and hyperlinks were among its characteristics. Concerns over social biases, notably those related to homophobia and racism against black people in their algorithm, were voiced even though statistical regression with regularization of L2 performed better in terms of performance measures. Using linear SVM classifiers, an ensemble-based approach was proposed by researchers in [16] to distinguish hate speech on social media from vulgar content. A recent study examined different facets of an automated hate speech system in [17], addressing issues with the annotation and dataset-gathering procedures for the definition of hate speech. Using word and character n-grams up to five as feature vectors, they created a nearly state-of-the-art multi-view stacking Support Vector Machine (mSVM) technique. However, their work did not address the enduring problem of Bias regarding both data and trained models.

Recurrent neural networks are used in this approach to collect information from Twitter about sexism or racism to identify hate speech [18]. Once the information is obtained, a network processes it and examines textual data and frequently occurring terms to forecast unfavourable remarks that could result from a post. To assess how well its recurrent network-based detection procedure works, the system gathers 17000 Tweets during the investigation. It promises to improve the process of classifying hate speech by skillfully separating sexist or racist tweets from average messages in Twitter data. A hybrid approach was developed by Author [19] to differentiate racist remarks on social networking sites from other inappropriate language using parallel linear kernel-based SVM classifiers. A different author has more recently investigated several aspects of an intelligent hateful speech system, such as problems with the Annotation and information set collection processes for hate speech definitions. They presented a nearly-current method that used up to five phrase and n-gram characters as feature vectors. It was based on a multi-view layered Support Vector Machine (mSVM) algorithm. Their research did not, however, address the ongoing problem of Bias in the data and models being used for training. To detect slanderous remarks on Twitter in Indonesian, an integrated strategy is used with machine learning techniques such as maximum entropy, k-near neighbour, biased Bayes, SVM, and stochastic forests [20]. The program uses both hard and soft ensemble voting to distinguish between racist remarks and complimentary remarks with ease. The system classifies the data from Twitter in Indonesia. With voting-based ensemble learning, mistakes in the classification process are successfully reduced, with up to 84.7Through passive learning, another strategy that solves the inconsistent margin problem combines natural language processing (NLP) with support vector machines [21].

TABLE I. LITERATURE SUMMARY

Ref	Problem	Method	Achievement	Limitations
[14]	Hate speech detection on Twitter	Manual annotation based on Critical Race Theory standards. Variable analysis with features like n-grams.	Effective classification into offensive and normal tweets. Emphasis on feature analysis.	Specific datasets and features limit generalizability.
[15]	Annotation and classification of offensive tweets on Twitter	Multiclass classifiers (RF, NB, DT, LR). Features include sentiment scores, POS labelling, n-grams, and TF-IDF.	Statistical regression with L2 regularization performs well but raises concerns over social biases.	Persistent biases, limited deep learning exploration.
[16]	Differentiating hate speech from vulgarity on social media	Ensemble-based method using SVM classifiers	Achieved state-of-the-art performance.	Bias in training data and models, limited contextual features.
[17]	Automated hate speech system	Phrase and n-gram characters as feature vectors in mSVM approach	Near state-of-the-art performance.	Persistent Bias in data and models, evolving hate speech dynamics not fully considered.
[18]	Hate speech detection on Twitter with RNN	Recurrent neural network processing textual information	Promises improvement in classifying hate speech.	Limited discussion on data collection biases.
[19]	Differentiating hate speech from abusive language	Ensemble-based approach with SVM classifiers	Achieved state-of-the-art performance.	Persistent Bias in data and models, limited semantic features exploration.
[20]	Hate speech recognition on Indonesian Twitter	Machine learning with ensemble voting	Successful classification with up to 84.7% accuracy.	Limited generalizability to other languages and cultures.
[21]	Addressing inconsistent margin problem in learning	Passive learning with SVMs	Improved computing efficiency.	Limited exploration of real-time processing.
[22], [23]	Predicting text using NLP models	LSTM and convolutional models	Effective prediction of text.	Limited discussion on training data biases.
[24], [25]	Deep learning in hate speech detection	ELMo, BERT, context-trained word vectors	Enhanced deep learning techniques.	Limited exploration of mid-end processing and training data biases.

Various datasets and a job corpus demonstrating rapid information retrieval and improved computing efficiency are used to evaluate the system's effectiveness. Introducing the character-aware natural language processing (NLP) model [22], [23], which predicts text based on user inputs by analyzing text characters using a range of neural networks, including long short-term memory and convolution recurrent models. Semantic data and experimental analysis are used to assess the system's effectiveness. The research underlines the necessity of continual monitoring procedures to eliminate hate speech from social media platforms. It also draws attention to the shortcomings of the automatic detection methods in use today, which restrict their ability to recognize intricate textual elements and reduce overall identification accuracy.

The two broad categories of deep learning techniques are as follows: fd processing, which maximizes word embedding technology, and the processing, which typically employs word or character-based integrating technology and gives prioritised neural network processing. ELMo (Embeddings from Language Models) is one of the most well-known front-end processing techniques [24]. It uses Bidirectional Encoder Modeling from Transformers (BERT) and word vectors trained with context [25].

III. Proposed System Model

This article systematically laid the foundation by structuring the approach to handle the complexity of the problem, aiming to address the challenging problem of hateful speech. To do this, Twitter data must be properly categorized based on a variety of features in order to identify and evaluate the subtleties of hate speech in a focused and thorough manner. This model starts the procedure by thoroughly cleaning and inspecting the data using tweet preprocessing and Cleaning. Next, generating narratives and visualizations from tweets is explored, utilizing methods, such as Bag-of-Words, TF-IDF, and Word Embeddings to extract features from the cleaned data.

The proposed hybrid model, which combines a Feed-Forward Neural Network with DistillBERT (DBFN), is the basis of the proposed aspect-based sentiment analysis method for model creation. Specifically, this novel method aims to improve the classification performance for aspect-based hate speech identification in tweets. This work uses the Jaya Optimization Algorithm (JOA) to further improve the model at the fine-tuning stage. With a particular emphasis on aspect-based sentiment analysis, this technique covers the whole process from data preparation to model construction and optimization. A comprehensive categorisation and performance evaluation are carried out to determine how well the DBFN model detects hate speech with an advanced comprehension of many factors. Fig. 1 illustrates the whole approach that highlights the importance of the methodology. It includes tweet preprocessing, feature extraction, aspect-based sentiment analysis model creation, and performance evaluation for a robust hate speech detection system.

A. Datasets Description

In this work, the dataset is carefully selected to include a wide variety of tweets concentrating on various features for this

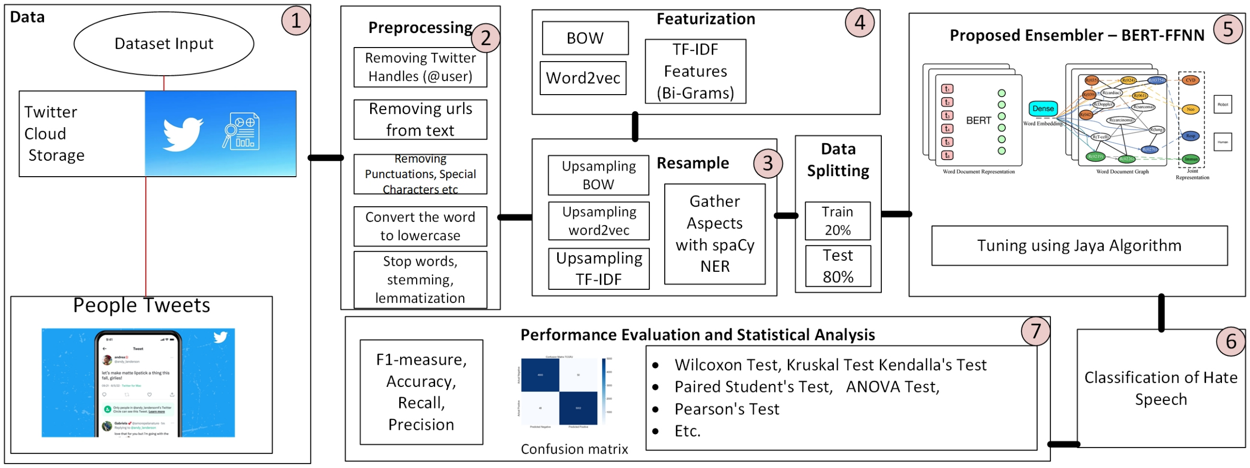
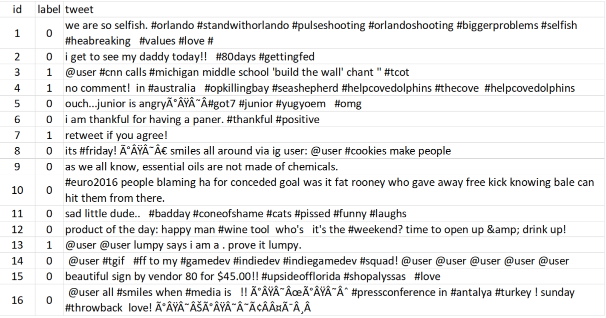

Data
1
Dataset Input
Twitter Cloud Storage
People Tweets
Preprocessing
2
Removing Twitter Handles (@user)
Removing urls from text
Removing Punctuations, Special Characters etc
Convert the word to lowercase
Stop words, stemming, lemmatization
Featurization
4
BOW
Word2vec
TF-IDF Features (Bi-Grams)
Resample
3
Upsampling BOW
Upsampling word2vec
Upsampling TF-IDF
Gather Aspects with spaCy NER
Data Splitting
Train 20%
Test 80%
Proposed Ensembler – BERT-FFNN
5
BERT
Dense
Word Embedding
Word Document Representation
Word Document Graph
Tuning using Jaya Algorithm
6
Classification of Hate Speech
Performance Evaluation and Statistical Analysis
7
F1-measure, Accuracy, Recall, Precision
Confusion matrix
Wilcoxon Test, Kruskal Test Kendalla's Test
Paired Student's Test, ANOVA Test,
Pearson's Test
Etc.

Fig. 1. Proposed framework for aspect based hate speech detection.

study on Aspect-Aware Hate Speech Detection in Tweets using a Hybrid of DistillBERT and Feed Forward Neural Network (DBFN). The data originates mostly from the repository at [26]. the structure of dataset is shown in Fig. 2.



id	label	tweet
1	0	we are so selfish. #orlando #standwithorlando #pulseshooting #orlandoshooting #biggerproblems #selfish #heabreaking #values #love #
2	0	i get to see my daddy today!! #80days #gettingfed
3	1	@user #cnn calls #michigan middle school 'build the wall' chant " #tcot
4	1	no comment! in #australia #opkillingbay #seashepherd #helpcovedolphins #thecove #helpcovedolphins
5	0	ouch...junior is angryÃ°ÂŸÂ˜Â#got7 #junior #yugyoem #omg
6	0	i am thankful for having a paner. #thankful #positive
7	1	retweet if you agree!
8	0	its #friday! Ã°ÂŸÂ˜Â€ smiles all around via ig user: @user #cookies make people
9	0	as we all know, essential oils are not made of chemicals.
10	0	#euro2016 people blaming ha for conceded goal was it fat rooney who gave away free kick knowing bale can hit them from there.
11	0	sad little dude.. #badday #coneofshame #cats #pissed #funny #laughs
12	0	product of the day: happy man #wine tool who's it's the #weekend? time to open up &amp; drink up!
13	1	@user @user lumpy says i am a . prove it lumpy.
14	0	@user #tgif #ff to my #gamedev #indiedev #indiegamedev #squad! @user @user @user @user @user
15	0	beautiful sign by vendor 80 for $45.00!! #upsideofflorida #shopalyssas #love
16	0	@user all #smiles when #media is !! Ã°ÂŸÂ˜ÂœÃ°ÂŸÂ˜Âˆ #pressconference in #antalya #turkey ! sunday #throwback love! Ã°ÂŸÂ˜ÂŠÃ°ÂŸÂ˜Â˜Ã¢ÂÂ¤Ã¯Â¸Â

Fig. 2. Unprocessed twitter dataset (tweets).

With this dataset, This work addresses distinct features of tweets for Aspect-Aware Hate Speech Detection. This technology can detect and evaluate subtleties in hate speech since aspect-wise data has been carefully curated.

B. Performing Preprocessing and EDA

This framework used a number of crucial procedures throughout the preparation stage for tweet texts to improve the consistency of the data. The process is shown in Fig. 3. First, Twitter handles (represented by "@user") were carefully eliminated using regular expressions to make sure that user-specific data didn't affect the study that followed. For example, a tweet that began "@user when a father is dysfunctional..." was changed to "when a father is dysfunctional ...".

Subsequently, hyperlinks and URLs present in the tweet's contents were eliminated by the use of regular expressions [27]. This can minimize noise from external connections and guarantee that the analysis entirely focuses on the textual content. The tweet, "Click and visit the link: http://example.com", was modified to say, "Click and visit the link:" Using regular expressions, the language was simplified by removing special characters, digits, and punctuation. This procedure aimed to remove superfluous symbols without affecting the information's meaning. This is now "in the mid-st century" instead of "in the mid-21st century ...".

After that, the tweet's content had lowercase versions of each word. This ensures consistency while reducing the information's dimensionality since "I Cannot Believe" is now simply "I cannot believe." Then, to concentrate on the tweets' more important substance, common stopwords like articles and prepositions were eliminated. For example, the sentence "when you know y'all 2 ain't going nowhere" was shortened to "know y'all 2 ain't going." Stemming was used to reduce terms to their root form and refine the data further. To merge related notions, the phrase "waiting for the show to start our third year running" becomes "wait for the show to start our third year run his technique."

The last technique used to contribute to a more advanced study is lemmatization, which reduces words to their dictionary or base form. One lemmatization of the phrase "waiting for the shows to start our third year running" was "waiting for the show to start our third year running". The Aspect-Aware Hate Speech Detection method uses the improved tweet text as a basis for further analysis, feature extraction, and model training after these preprocessing processes are completed. The sample of preprocessed tweets is shown in Fig. 4.

C. Featurization and Resampling

In the featurization and resampling phase, the objective is to convert preprocessed tweet text into numerical features using Bag-of-Words (BOW), TF-IDF Features with Bi-Grams, and Word2Vec embeddings [28].

a) Bag-of-Words (BOW):: The model used in this study employed the CountVectorizer function to transform the text data into a matrix of token counts [29]:

$$\mathrm{df\_bow} = \mathrm{CountVectorizer}(\mathrm{stop\_words}{=}\mathrm{english}).\mathrm{fit\_transform}(\mathrm{text\_data}) \qquad (1)$$

This process captures the frequency of each word in the text, providing a numerical representation for subsequent analysis.

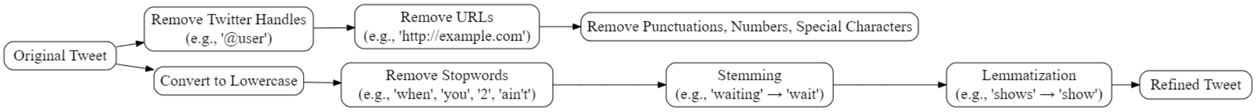
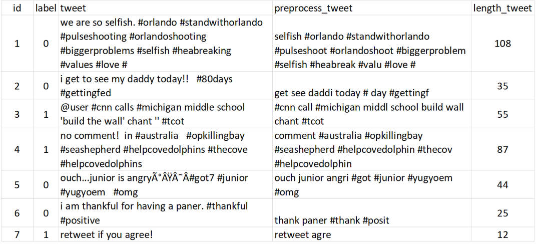

Original Tweet
Remove Twitter Handles (e.g., '@user')
Remove URLs (e.g., 'http://example.com')
Remove Punctuations, Numbers, Special Characters
Convert to Lowercase
Remove Stopwords (e.g., 'when', 'you', '2', 'ain't')
Stemming (e.g., 'waiting' → 'wait')
Lemmatization (e.g., 'shows' → 'show')
Refined Tweet

Fig. 3. Process of preprocessing.



id	label	tweet	preprocess_tweet	length_tweet
1	0	we are so selfish. #orlando #standwithorlando #pulseshooting #orlandoshooting #biggerproblems #selfish #heabreaking #values #love #	selfish #orlando #standwithorlando #pulseshoot #orlandoshoot #biggerproblem #selfish #heabreak #valu #love #	108
2	0	i get to see my daddy today!! #80days #gettingfed	get see daddi today # day #gettingf	35
3	1	@user #cnn calls #michigan middle school 'build the wall' chant '' #tcot	#cnn call #michigan middl school build wall chant #tcot	55
4	1	no comment! in #australia #opkillingbay #seashepherd #helpcovedolphins #thecove #helpcovedolphins	comment #australia #opkillingbay #seashepherd #helpcovedolphin #thecov #helpcovedolphin	87
5	0	ouch...junior is angryÃ°ÂŸÂ˜Â#got7 #junior #yugyoem #omg	ouch junior angri #got #junior #yugyoem #omg	44
6	0	i am thankful for having a paner. #thankful #positive	thank paner #thank #posit	25
7	1	retweet if you agree!	retweet agre	12

Fig. 4. Preprocessed tweets.

b) TF-IDF Features with Bi-Grams:: Utilizing the *TfidfVectorizer* function, the equation for TF-IDF with Bi-Grams is given by [30]:

$$\text{df\_tfidf} = \text{TfidfVectorizer}(\text{ngram\_range=(1, 2), stop\_words='english'}).\text{fit\_transform}(\text{text\_data}) \tag{2}$$

This technique considers the importance of terms by incorporating individual words and two-word phrases.

c) Word2Vec: : The `Word2Vec` function was used to create Word2Vec embeddings [30]:

$$\begin{aligned}\text{Word2Vec} = \text{df\_w2v}(\text{window} = 5, \text{sentences},\\ \text{workers} = 4, \text{min\_count} = 1, \text{vector\_size} = 100)\end{aligned} \tag{3}$$

Word2Vec captures the semantic links between words and provides a detailed depiction of the underlying semantics in the tweet text.

d) Resampling Techniques: : To address class imbalance, the model implemented resampling techniques on the datasets:

Upsampling BOW: To match the majority class (label 0), upsampling entails boosting the occurrences of the minority class (label 1) within the BOW dataset. The equation for upsampling BOW is [31]:

$$\text{df\_bow\_upsampled} = \text{resample}(\text{df\_minor, replace=True, n\_samples=major\_class\_0}) \tag{4}$$

This technique ensures a balanced representation of both classes in the training data.

Upsampling TF-IDF: Similar to BOW, the TF-IDF dataset underwent upsampling to achieve a balanced class distribution. Eq. 5 show the Tf-IDF upsampling [32]:

$$\text{df\_tfidf\_upsampled} = \text{resample}(\text{df\_minor, replace=True, n\_samples=major\_class\_0}) \tag{5}$$

Upsampling helps prevent biases towards the majority class.

Upsampling Word2Vec: The Word2Vec dataset was upsampled to address the class imbalance. The equation for upsampling Word2Vec is [32]:

$$\text{df\_w2v\_upsampled} = \text{resample}(\text{df\_minor, replace=True, n\_samples=major\_class\_0}) \tag{6}$$

This technique ensures a fair representation of both classes in the training data.

In the proposed Aspect-Aware Hate Speech Detection system, these resampling strategies are essential for avoiding biases, improving model performance, and preserving an equal proportion of non-hate speech and hate speech occurrences. Visualizations, such as count plots, were generated to illustrate the balanced class distribution in the upsampled datasets.

D. Proposed DBFN-SHO

The architecture and design of the suggested aspect-aware hate speech detection model called the DistillBERT [33] and Feed Forward Neural Network [34] (DBFN) are presented in this section. The DBFN model is a hybrid system that effectively classifies hate speech in tweets by combining the strength of a feed-forward neural network with DistillBERT, a simplified version of BERT (Bidirectional Encoder Representations from Transformers).

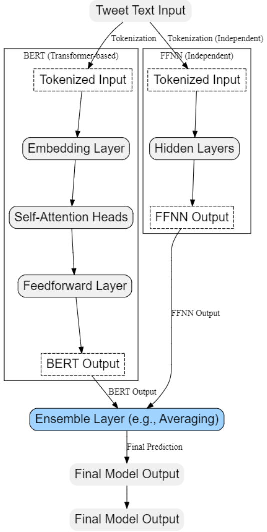


Fig. 5. Proposed DBFN model.

The transformer-based model DistillBERT, represented as DB, extracts word contextual embeddings to produce a contextualized representation of the input text. Parameterizing the model is done with θ_{DB} [34].

By modifying the parameters θ_{DB} of DistillBERT, the model is optimized for the hate speech dataset to reduce the cross-entropy loss [33], [34]:

$$\mathcal{L}_{\text{DB}}(\theta_{\text{DB}}) = -\frac{1}{N}\sum_{i=1}^{N}\left[y_i \log(P_{\text{DB}}(x_i;\theta_{\text{DB}})) + (1-y_i)\log(1-P_{\text{DB}}(x_i;\theta_{\text{DB}}))\right] \tag{7}$$

1) Feed Forward Neural Network: FFNN is a classifier defined by θ_{FFNN}. This is also known as the Feed-Forward Neural Network. The projected likelihood of hate speech is output, and the contextual embeddings generated by DistillBERT are used as input [34].

$$\hat{y} = P_{\text{FFNN}}(P_{\text{DB}}(x;\theta_{\text{DB}});\theta_{\text{FFNN}}) \tag{8}$$

a) Training and Optimization: The cross-entropy loss is minimized to optimize the parameters θ_{FFNN}:

$$\mathcal{L}_{\text{FFNN}}(\theta_{\text{FFNN}}) = -\frac{1}{N}\sum_{i=1}^{N}[y_i \log(\hat{y}_i) + (1-y_i)\log(1-\hat{y}_i)] \tag{9}$$

b) Aspect-Aware Classification: The DBFN model includes an aspect-aware categorization technique called Aspect-Aware [35] that takes into account many aspects seen in tweets that contain hate speech. The set of aspects, such as gender, race, and religion, is represented by A. The function $P_{\text{Aspect-Aware}}(x)$ generates the aspect-aware predictions for tweet x.

$$P_{\text{Aspect-Aware}}(x) = P_{\text{AA}}(P_{\text{DB}}(x;\theta_{\text{DB}}), P_{\text{FFNN}}(P_{\text{DB}}(x;\theta_{\text{DB}});\theta_{\text{FFNN}});\theta_{\text{AA}}) \tag{10}$$

This work investigates ensemble learning strategies to further improve the DBFN model's resilience by embedding the FFNN inside the BERT and, in the end, by merging predictions gathered from various Feed Forward Neural Network and DistillBERT instances, as shown in Fig. 5.

$$P_{\text{Ensemble}}(x) = \frac{1}{K}\sum_{k=1}^{K} P_{\text{AA}_k}(x) \tag{11}$$

E. Parameter Tuning with Jaya Optimization Algorithm (JOA)

Aspect-Aware Hate Speech Detection model DBFN performs best when hyperparameters are fine-tuned with suggested technique JOA [36]. JOA is a method motivated by cooperative population dynamics. Optimization is applied to the following hyperparameters: epoch count, learning rate, batch size, DB hidden units, and FFNN hidden units. These hyperparameters are essential factors that affect the model's accuracy, reliability, and stability. Table II summarises the optimum values for each hyperparameter.

TABLE II. OPTIMISTIC HYPERPARAMETERS AND THEIR APPROPRIATE VALUES FOR TUNING

Hyperparameter	Optimized Value
Batch Size:	128
DistillBERT Hidden Units	256
FFNN Hidden Units	128
Epochs	30
Learning Rate	0.0005

The JOA is fully defined in Algorithm 1 [36]. Iteratively adjusts hyperparameter settings based on the model's efficacy on a validation set. The algorithm investigates the hyperparameter space and modifies configurations using crossover and mutation procedures. Until convergence is reached, the process keeps going.

Algorithm 1 Jaya Optimization Algorithm for Hyperparameter Tuning

```
Input:
a collection of hyperparameter settings $I$
$f(\text{configuration})theobjectivefunction.$
Range for every $[L, U]$ base_parameter STATE threshold of convergence $\theta$
Output: Optimimal values of base_parameters
Optimization_Jaya
The convergence threshold $\theta$
Set up the optimal arrangement first: $y_{\text{best}}$ from $I$
while values Not meet do
    for Every $y_i$ configuration in $P$ do
        Using $e$, consistently generate a random number in the interval $[0, 1]$.
        Revise the setup:
            $y_j = r \cdot (y_{\text{best}} - y_j) + y_j$
        Verify that the parameters are within the specified range.
            $y_j = \min(\max(y_j, L), U)$
    end for
    Determine which configuration, $y_{\text{best}}$, has the highest value of the objective function.
end while
```

The proposed hate speech detection model, DBFN-J, is more predictive and resilient when the Jaya Optimization Algorithm is included. This refined model, which encapsulates the DB-based hybrid architecture, makes effectively detecting and categorizing hate speech elements in tweets possible.

Using data preprocessing and the Jaya-optimized DBFN model training, Algorithm 1 provides a comprehensive overview of the proposed hate speech detection model. The resultant model, DBFN-J, is intended to offer accurate and trustworthy predictions for aspect-aware hate speech detection within the context of tweets on social media.

F. Classification Assessment Metrics

This methodology uses a hybrid DBFN approach to identify the features of hate speech in tweets. It implements numerous parameters for assessing the efficacy of the proposed technique and verifying its accuracy and usefulness in detecting hate speech in different situations [37].

a) AUC and ROC Analysis: The Receiver Operating Characteristic Curve (ROC) metrics are applied to determine whether a precise approach is effective in recognizing various aspects of hate speech. The curved shape depicts the disparity between realistic positive effects and incorrect negative results. The model's overall discriminative ability is evaluated using the Area Under the Curve (AUC) [38]. TPR measures the capacity of the algorithm to recognize inappropriate speech through its attributes. However, FPR measures the technique's capacity to identify the difference between hate speech and non-hateful content. The study of ROC curves and AUC estimation is implemented to accurately assess the method's effectiveness in recognizing hateful speech.

b) Accuracy and Recall: The proactive stability and durability of the framework towards understanding hate speech features will be assessed by applying specific metrics. The algorithm's accuracy evaluates its ability to identify hate speech characteristics, particularly in challenging circumstances. The recall comparisons determine how effectively the model differentiates hate speech in practical situations and excludes instances, and it performs inadequately in this aspect. Eq. 12 [36] and 13 [37] will be used to assess the recall and accuracy using the parameters of the TPR, FPR, and TNR.

$$\text{Precision} = \frac{\text{TPR}}{\text{TPR} + \text{FPR}} \tag{12}$$

$$\text{Recall} = \frac{\text{TPR}}{\text{TPR} + \text{FNR}} \tag{13}$$

By indicating the anticipated reliability of the technique, these variables yield essential details concerning the degree to which the technology recognizes aspects of hate speech.

c) Logloss Assessment:: The corresponding decrease in exponential accuracy is a significant statistic that matters when measuring the predicted efficiency of a hate speech recognition strategy. This metric is illustrated by the coefficient estimator 14 [38], which determines the disparity between the estimated chances and actual probability.

$$\log \text{Loss} = -\frac{1}{M} \sum_{j=1}^{M} \left(x_i \log(I_j) + (1 - x_j) \log(1 - I_j)\right) \tag{14}$$

This algorithm assesses the model's fit across its likelihood estimates and the true identifiers. Highlighting an increased correspondence with the predicted and actual labels, a decrease in log deficit implies an improvement in recognizing hate speech aspects.

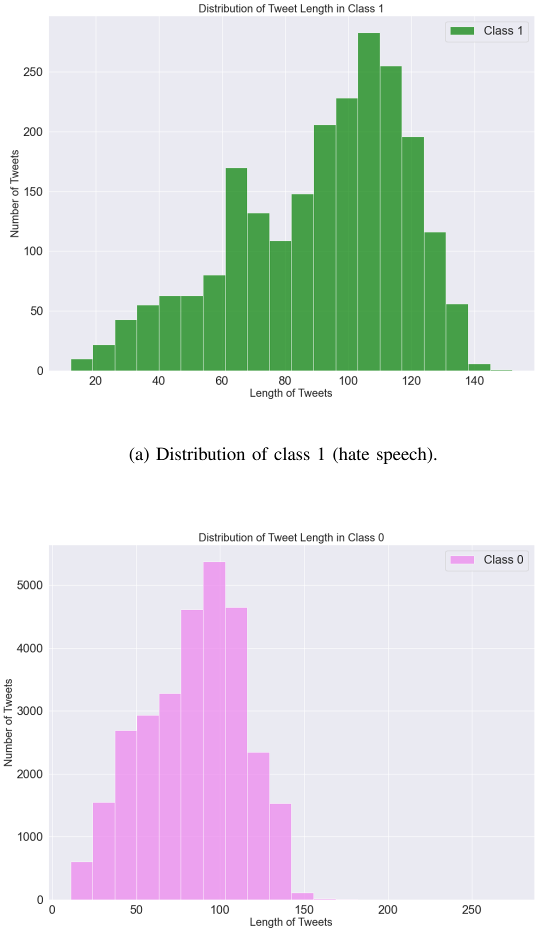
Statistical Analysis for Assessment: A rigorous statistical assessment is employed to assess the combined DBFN technique using other strategies and basic models. Several statistical approaches, such as ANOVA, Student's t-test, median deviation, standard deviation, and range, are applied throughout the evaluation to assess the variety and value of the data. Researchers analyze the computational difficulty to estimate the additional resources needed to facilitate the hybrid approach's implementation. The in-depth examination proposes an extensive explanation of the adaptation and usability of the suggested approach.

The hybrid strategy approach aims to recognize features that characterize hate speech in tweets. It offers in-depth knowledge of each computational efficacy and accurate prediction reliability. Comprehensive statistical studies and the previously outlined assessment criteria, which produce significant data, demonstrate the model's predicted accuracy in many aspects of hate speech.

IV. Simulation Results and Discussion

The proposed framework is simulated on a computer with a Core i7 processor and 32GB RAM. This study was carried out using Python. Multiple datasets containing tweets were combined for analysis related to hate speech. The choice of this dataset is attributed to its updated status in 2022, and extensive simulations were conducted to assess its effectiveness and flexibility. The experimental results are elaborated upon in the subsequent discussion.

Initially, the hate speech dataset was investigated. A detailed summary of the dataset's noteworthy technical and demographic trends can be found in Fig. 6. The histograms provide insightful information about how tweet durations are distributed throughout the dataset's various classifications. Fig. 6a's histogram depicts the length distribution of Class 1 tweets or tweets containing hate speech. The green bars, which display the frequency of tweets at various durations, provide a thorough understanding of the distribution pattern within this class. A similar analysis is demonstrated for tweets in Class 0 (non-hate speech) in the Fig. 6b histogram.



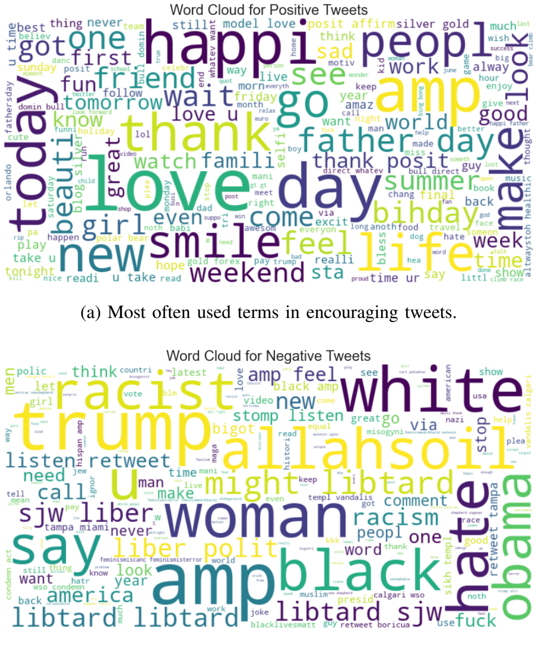
(a) Distribution of class 1 (hate speech).

(b) Distribution of Class 0 (non-hate speech)

Fig. 6. Comparison of tweet length distributions.

Table III shows the technical specs of hate speech detection models that use the proposed DBFN-J model and their baseline versions. With a low log loss of 0.06, high accuracy of 0.97, and intense discrimination, as evidenced by an AUC of 0.989, the DBFN-J model performs better than the others. Metrics such as ROC-CH (0.95) and MCC

(0.93) highlight its effectiveness, while an F1-Score of 0.95 is the result of impressively balanced accuracy (0.98) and recall (0.97). When compared to baseline models, DBFN-J routinely performs better or on par with them, highlighting its superiority in hate speech identification through precise predictions and sophisticated discrimination. This suggests that DBFN-J is a viable solution to the widespread problem of hate speech on the internet.

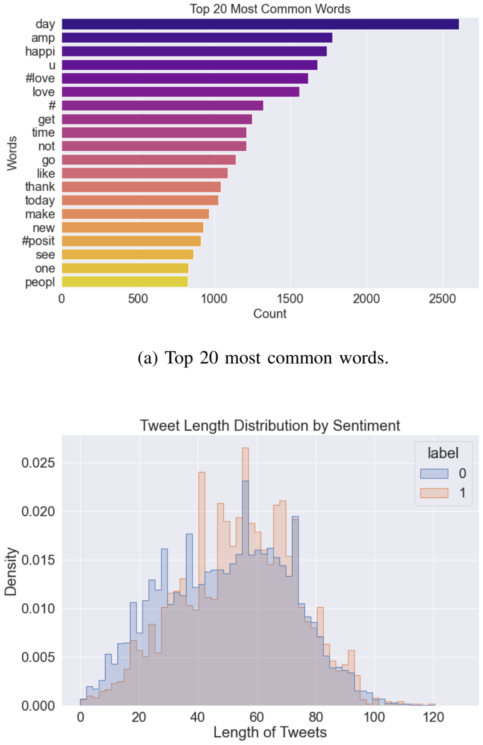


(a) Most often used terms in encouraging tweets.

(b) Most often used terms in negative tweets.

Fig. 7. Positive and negative tweets.

The most common terms in encouraging tweets are represented graphically in this word cloud Fig. 7. The magnitude of each word in the positive tweets relates to how frequently it appears, as shown in Fig. 7a. A word occurs more often when it is more extensive. Understanding the recurring themes or encouraging feelings stated in the dataset may be gained via analyzing this word cloud. For example, the prominence of adjectives such as joyful, fantastic, and love suggests that positive feelings are prevalent in the positive class. The word clouds in Fig. 7b show which terms are frequently used in unfavourable tweets. Every word's magnitude reflects how often it appears in nasty tweets. Examining this word cloud might provide information about the negative themes or attitudes that are most common in the dataset. Words like "hatred," "sad," or "angry" may be often used, showing that the negative class has prevalent negative attitudes.



(a) Top 20 most common words.

(b) Tweet length distribution by sentiment.

Fig. 8. Common words and tweet length distribution.

Fig. 8 shows the data distribution with another angle. The top 20 most often occurring terms throughout the whole dataset are shown in Fig. 8a. The bar graph shows how frequently words appear and how various words are distinguished by their colour intensity. This study aids in determining the most commonly used phrases in tweets. For example, the dataset's prevalence of abusive versus non-abusive terms may reveal the prevailing language patterns. The sentiment-based distribution of tweet durations distinguishing between offensive and non-abusive tweets is shown in Fig. 8b. Histograms are used in this graphic to display the density of tweet size for each sentiment class. It helps us understand the distribution and range of tweet durations in both groups. Examples of possible observations are whether a class tends to tweet longer or shorter or whether there are overlapping zones. Trends in tweet length and mood may be found using this data.

TABLE III. PERFORMANCE EVALUATION OF CLASSIFICATION MODELS

Model	Log Loss	Accuracy	AUC	Precision	F1-Score	Recall	ROC-CH	MCC
DBFN-J (Proposed)	0.06	0.97	0.99	0.98	0.95	0.97	0.95	0.93
BERT [19]	0.73	0.73	0.79	0.66	0.75	0.86	0.72	0.61
RNN[18]	0.42	0.87	0.85	0.88	0.83	0.87	0.82	0.79
CNN-LSTM [22]	0.59	0.77	0.7	0.75	0.79	0.84	0.68	0.65
SVM [16]	0.42	0.82	0.76	0.86	0.84	0.83	0.74	0.72
NB [15]	0.39	0.9	0.87	0.94	0.92	0.9	0.88	0.86
KNN [20]	0.2	0.847	0.88	0.79	0.86	0.9	0.85	0.82
ELMo	0.42	0.83	0.8	0.87	0.89	0.88	0.79	0.76
BERT-LSTM	0.1	0.92	0.90	0.91	0.89	0.928	0.89	0.87

The DBFN-J model that has been suggested performs better technically than the other models that have been assessed in Table III. The model displays remarkable accuracy and makes exact predictions with a shallow log loss of 0.06. Its robust AUC of 0.99 indicates that the algorithm can efficiently identify between offensive and non-abusive comments. Despite its anticipated positive effects, the approach yields a remarkable precision of 0.98, exhibiting high text abuse recognition efficiency. Such an equity F1-score of 0.95 indicates the algorithm's model can integrate recall and accuracy. However, a recall of 0.97 demonstrates that the procedure may differentiate between aggressive and favourable tweets.

The ROC-CH of 0.95 for this framework demonstrates its remarkable capacity to balance TP and FP rates. In addition, an MCC value of 0.93 displays the algorithm's general efficiency and indicates significant consistency between the estimated and observed categorization. The combination of scientific findings indicates the reliability of the DBFN-J framework for identifying inappropriate comments.

TABLE IV. AVERAGE COMPUTATIONAL TIME ANALYSIS

Model	Median (s)	Mean (s)	Min (s)	Max (s)	Range (s)	Std. Dev. (s)
BERT [19]	86	87	75	97	21	5.21
CNN [17]	85	85	74	96	21	5.36
BERT-LSTM [17]	86	87	78	96	17	4.15
CNN-LSTM [21]	86	89	73	96	22	5.39
ELMo [27]	85	84	75	97	21	4.95
SVM [19]	84	84	76	92	15	4.09
DBFN-J (Proposed)	36	35	33	40	4	0.11

Table IV contains a discussion of the processing time that indicates the various gains among various study methods. Compared to the remainder models, the DBFN-J approach is more efficient, as evidenced by its substantially quicker median and mean execution times and less uncertainty. This demonstrates the highly computationally efficient suggested model, making it a good option for real-world scenarios where processing speed is crucial. Researchers compare the mean, median, and standard deviation of the various models—including BERT, CNN, BERT-LSTM, ResNet, ELMo, and SVM—to thoroughly understand each model's efficiency profile. These models display differing degrees of computing performance.

TABLE V. STATISTICAL ANALYSIS OF THE PROPOSED WRNG-J AND EXISTING METHODS

Method	Test	F-stat	P-Value
BERT	Kendall's	0.553	-0.011
	Pearson's	0.623	-0.011
	Chi-Squared	105.21	0.004
	Spearman's	0.598	-0.011
ELMo	Kendall's	0.623	-0.011
	Pearson's	0.623	-0.011
	Chi-Squared	101.694	-0.011
	Spearman's	0.623	-0.011
DBFN-J	Kendall's	0.79	-0.011
	Pearson's	0.888	-0.011
	Chi-Squared	109.429	0.042
	Spearman's	0.856	-0.011
CNN-LSTM	Kendall's	0.623	-0.011
	Pearson's	0.623	-0.011
	Chi-Squared	101.694	-0.011
	Spearman's	0.623	-0.011
Non-Parametric Tests	Wilcoxon	15313.989	0.156
	Kruskal	6.706	0.008
	Mann-Whitney	26519.989	-0.011
Parametric Tests	Student's	-0.742	0.454
	Paired Student's	-1.079	0.285
	ANOVA	0.533	0.454

After a detailed statistical study, Table V highlights the crucial trends and distinctions between the effectiveness of the recommended and current strategies. In addition to providing an extensive data breakdown and significant levels for many statistical tests, the table enables a comprehensive evaluation of the advantages and disadvantages of the different approaches. A "0" p-value indicates the absence of statistically significant impacts or differences. From a statistical perspective, a result is considered vital if it means a difference or impact and has a p-value more than zero, which is still highly tiny (preferably less than 0.04). The p-value in this instance is insignificant because p-values are typically positive. As a result, care must be used while examining data with negative p-values.

V. CONCLUSION AND FUTURE DIRECTIONS

The hate speech recognition system driven by the DBFN-J model is a significant development in the field. Using a large-scale Twitter dataset collected over the previous four years from a GitHub repository, the system uses NLP tokenization to do careful data preprocessed. The dataset is better when unnecessary elements, such as data characters, hashtags, and user information, are removed. By investigating semantic sentiment unigram and pattern characteristics, the system derives insightful information and creates vectors that guide further categorization. An ensemble of deep neural network classifiers enhanced by adding the Jaya method for fine-tuning parameters performs remarkably well. With a good accuracy rate of 97% and a small loss function of 0.06, the DBFN-J model demonstrates its effectiveness in identifying hate speech. This work is noteworthy for its lightweight and efficient technique, which outperforms well-established models like CNN and BERT ELMo in terms of performance. The application of hybrid techniques further strengthens the total classification accuracy.

Although effective, the DBFN-J model has drawbacks. One dataset from Twitter may limit the model's applicability. Second, while effective, preprocessing may remove hashtags and user metadata, affecting model interpretability. The model may also struggle to identify subtle or implicit hate speech in low-resource languages. Future research can use multimodal social media datasets to improve model adaptability and robustness. Future research may improve implicit hate speech detection with transformer-based architectures like GPT or T5. Multilingual models for low-resource languages and cultural differences are promising. Finally, real-time deployment of the DBFN-J model with dynamic feedback mechanisms for continuous improvement may help combat online hate speech.

ACKNOWLEDGMENT

The work was funded by the University of Jeddah, Jeddah, Saudi Arabia, under grant number (UJ-23-SHR-77). The authors, therefore, acknowledge with thanks the University of Jeddah for technical and financial support.

REFERENCES

[1] S. Windisch, S. Wiedlitzka, A. Olaghere, and E. Jenaway, *Online interventions for reducing hate speech and cyberhate: A systematic review*, Campbell Systematic Reviews, vol. 18, no. 2, pp. e1243, 2022.

[2] A. Tontodimamma, E. Nissi, A. Sarra, and L. Fontanella, *Thirty years of research into hate speech: topics of interest and their evolution*, Scientometrics, vol. 126, no. 1, pp. 157–179, 2021.

[3] E. Aswad and D. Kaye, *Convergence & conflict: reflections on global and regional human rights standards on hate speech*, Nw. UJ Int'l Hum. Rts., vol. 20, pp. 165, 2021.

[4] B. Nyagadza, *Search engine marketing and social media marketing predictive trends*, Journal of Digital Media & Policy, vol. 13, no. 3, pp. 407–425, 2022.

[5] R. Qasim, W. H. Bangyal, M. A. Alqarni, and A. A. Almazroi, *A fine-tuned BERT-based transfer learning approach for text classification*, Journal of Healthcare Engineering, vol. 2022, no. 1, pp. 1–11, 2022.

[6] H. Simon, B. Y. Baha, and E. J. Garba, *Trends in machine learning on automatic detection of hate speech on social media platforms: A systematic review*, FUW Trends in Science & Technology Journal, vol. 7, no. 1, pp. 001–016, 2022.

[7] A. A. Almazroi, L. Abualigah, M. A. Alqarni, E. H. Houssein, A. Q. M. AlHamad, and M. A. Elaziz, *Class Diagram Generation from Text Requirements: An Application of Natural Language Processing*, in Deep Learning Approaches for Spoken and Natural Language Processing, Springer, pp. 55–79, 2021.

[8] K. Sharifani and M. Amini, *Machine learning and deep learning: A review of methods and applications*, World Information Technology and Engineering Journal, vol. 10, no. 07, pp. 3897–3904, 2023.

[9] A. A. Almazroi, *A fast hybrid algorithm approach for the exact string matching problem via berry ravindran and alpha skip search algorithms*, Journal of Computer Science, vol. 7, no. 5, pp. 644, 2011.

[10] R. T. Mutanga, N. Naicker, and O. O. Olugbara, *Detecting Hate Speech on Twitter Network using Ensemble Machine Learning*, International Journal of Advanced Computer Science and Applications, vol. 13, no. 3, pp. 1–10, 2022.

[11] B. Kennedy, M. Atari, A. M. Davani, L. Yeh, A. Omrani, Y. Kim, and M. Dehghani, *Introducing the Gab Hate Corpus: defining and applying hate-based rhetoric to social media posts at scale*, Language Resources and Evaluation, pp. 1–30, 2022.

[12] P. Chiril, E. W. Pamungkas, F. Benamara, V. Moriceau, and V. Patti, *Emotionally informed hate speech detection: a multi-target perspective*, Cognitive Computation, pp. 1–31, 2022.

[13] M. Wankhade, A. C. S. Rao, and C. Kulkarni, *A survey on sentiment analysis methods, applications, and challenges*, Artificial Intelligence Review, vol. 55, no. 7, pp. 5731–5780, 2022.

[14] S. Nagar, F. A. Barbhuiya, and K. Dey, *Towards more robust hate speech detection: using social context and user data*, Social Network Analysis and Mining, vol. 13, no. 1, pp. 47, 2023.

[15] P. Som, R. Mishra, S. Das, R. K. Singh, D. K. Rakesh, B. Behera, and R. R. Kumar, *Evaluating Machine Learning Models for Hate Speech Detection in ODIA Language*, in 2024 1st International Conference on Cognitive, Green and Ubiquitous Computing (IC-CGU), IEEE, pp. 1–6, 2024.

[16] A. C. Mazari, N. Boudoukhani, and A. Djeffal, *BERT-based ensemble learning for multi-aspect hate speech detection*, Cluster Computing, vol. 27, no. 1, pp. 325–339, 2024.

[17] S. Chinivar, M. S. Roopa, J. S. Arunalatha, and K. R. Venugopal, *Online offensive behaviour in social media: Detection approaches, comprehensive review and future directions*, Entertainment Computing, vol. 45, pp. 100544, 2023.

[18] K. Maity, G. Balaji, and S. Saha, *Towards Analyzing the Efficacy of Multi-task Learning in Hate Speech Detection*, in International Conference on Neural Information Processing, Springer Nature Singapore, pp. 317–328, 2023.

[19] H. Saleh, A. Alhothali, and K. Moria, *Detection of hate speech using BERT and hate speech word embedding with deep model*, Applied Artificial Intelligence, vol. 37, no. 1, pp. 2166719, 2023.

[20] I. P. Sari and H. Maulana, *Detecting Cyberbullying on Social Media Using Support Vector Machine: A Case Study on Twitter*, International Journal of Safety & Security Engineering, vol. 13, no. 4, pp. 1–10, 2023.

[21] S. Saifullah, R. Dreżewski, F. A. Dwiyanto, A. S. Aribowo, Y. Fauziah, and N. H. Cahyana, *Automated text annotation using a semi-supervised approach with meta vectorizer and machine learning algorithms for hate speech detection*, Applied Sciences, vol. 14, no. 3, pp. 1078, 2024.

[22] H. Vanam and J. R. Raj, *CNN-OLSTM: Convolutional Neural Network with Optimized Long Short-Term Memory Model for Twitter-based Sentiment Analysis*, IETE Journal of Research, pp. 1–12, 2023.

[23] S. Zhong, A. Scarinci, and A. Cicirello, *Natural language processing for systems engineering: automatic generation of systems modelling language diagrams*, Knowledge-Based Systems, vol. 259, pp. 110071, 2023.

[24] A. Kumar and S. Kumar, *Hate speech detection in multi-social media using deep learning*, in International Conference on Advanced Communication and Intelligent Systems, Springer Nature Switzerland, pp. 59–70, 2023.

[25] A. Balayn, J. Yang, Z. Szlavik, and A. Bozzon, *Automatic identification of harmful, aggressive, abusive, and offensive language on the web: A survey of technical biases informed by psychology literature*, ACM Transactions on Social Computing (TSC), vol. 4, no. 3, pp. 1–56, 2021.

[26] A. Ammar, *Datasets for Hate Speech Detection*, Retrieved from https://github.com/aymeam/Datasets-for-Hate-Speech-Detection.

[27] F. Fkih, T. Moulahi, and A. Alabdulatif, *Machine learning model for offensive speech detection in online social networks slang content*, WSEAS Trans. Inf. Sci. Appl., vol. 20, pp. 7–15, 2023.

[28] H. A. Madni, M. Umer, N. Abuzinadah, Y. C. Hu, O. Saidani, S. Alsubai, M. Hamdi, and I. Ashraf, *Improving sentiment prediction of textual tweets using feature fusion and deep machine ensemble model*, Electronics, vol. 12, no. 6, pp. 1302, 2023.

[29] S. Dai, K. Li, Z. Luo, P. Zhao, B. Hong, A. Zhu, and J. Liu, *AI-based NLP section discusses the application and effect of bag-of-words models and TF-IDF in NLP tasks*, Journal of Artificial Intelligence General Science (JAIGS), vol. 5, no. 1, pp. 13–21, 2024.

[30] A. Dey, M. Jenamani, and J. J. Thakkar, *Lexical TF-IDF: An n-gram feature space for cross-domain classification of sentiment reviews*, in International Conference on Pattern Recognition and Machine Intelligence, Springer, pp. 380–386, 2017.

[31] D. Rau, M. Dehghani, and J. Kamps, *Revisiting Bag of Words Document Representations for Efficient Ranking with Transformers*, ACM Transactions on Information Systems, vol. 42, no. 5, pp. 1–27, 2024.

[32] A. Banerjee, P. Shivakumara, S. Bhattacharya, U. Pal, and C. L. Liu, *An end-to-end model for multi-view scene text recognition*, Pattern Recognition, vol. 149, pp. 110206, 2024.

[33] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, *DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter*, arXiv preprint arXiv:1910.01108, 2019.

[34] H. Saleh, A. Alhothali, and K. Moria, *Detection of hate speech using BERT and hate speech word embedding with deep model*, Applied Artificial Intelligence, vol. 37, no. 1, pp. 2166719, 2023.

[35] N. Ayub, H. Tayyaba, S. Hussain, S. S. Ullah, and J. Iqbal, *An Efficient Optimized DenseNet Model for Aspect-Based Multi-Label Classification*, Algorithms, vol. 16, no. 12, pp. 548, 2023.

[36] R. A. Zitar, M. A. Al-Betar, M. A. Awadallah, I. A. Doush, and K. Assaleh, *An intensive and comprehensive overview of JAYA algorithm, its versions and applications*, Archives of Computational Methods in Engineering, vol. 29, no. 2, pp. 763–792, 2022.

[37] A. A. Almazroi, O. A. Mohamed, A. Shamim, and M. Ahsan, *Evaluation of State-of-the-Art Classifiers: A Comparative Study*, Journal of Computing, vol. 1, no. 1, pp. 22–29, 2020.

[38] D. Chicco and G. Jurman, *The Matthews correlation coefficient (MCC) should replace the ROC AUC as the standard metric for assessing binary classification*, BioData Mining, vol. 16, no. 1, pp. 4, 2023.