applied sciences

MDPI

Article

Enhanced Seagull Optimization with Natural Language Processing Based Hate Speech Detection and Classification

Yousef Asiri [1], Hanan T. Halawani [1], Hanan M. Alghamdi [2], Saadia Hassan Abdalaha Hamza [3], Sayed Abdel-Khalek [4,5,*] and Romany F. Mansour [6]

1 Department of Computer Science, College of Computer Science and Information Systems, Najran University, Najran 61441, Saudi Arabia
2 Department of Computer Science, College of Computing Al Qunfidhah, Umm Al-Qura University, Mecca 24382, Saudi Arabia
3 Department of Computer Science, College of Science and Humanities, Prince Sattam Bin AbdulAziz University, Slayel 11913, Saudi Arabia
4 Department of Mathematics, Faculty of Science, Sohag University, Sohag 82524, Egypt
5 Department of Mathematics, College of Science, Taif University, Taif 21944, Saudi Arabia
6 Department of Mathematics, Faculty of Science, New Valley University, El-Kharga 72511, Egypt
* Correspondence: sabotalb@tu.edu.sa

Abstract: Hate speech has become a hot research topic in the area of natural language processing (NLP) due to the tremendous increase in the usage of social media platforms like Instagram, Twitter, Facebook, etc. The facelessness and flexibility provided through the Internet have made it easier for people to interact aggressively. Furthermore, the massive quantity of increasing hate speech on social media with heterogeneous sources makes it a challenging task. With this motivation, this study presents an Enhanced Seagull Optimization with Natural Language Processing Based Hate Speech Detection and Classification (ESGONLP-HSC) model. The major intention of the presented ESGONLP-HSC model is to identify and classify the occurrence of hate speech on social media websites. To accomplish this, the presented ESGONLP-HSC model involves data pre-processing at several stages, such as tokenization, vectorization, etc. Additionally, the Glove technique is applied for the feature extraction process. In addition, an attention-based bidirectional long short-term memory (ABLSTM) model is utilized for the classification of social media text into three classes such as neutral, offensive, and hate language. Moreover, the ESGO algorithm is utilized as a hyperparameter optimizer to adjust the hyperparameters related to the ABLSTM model, which shows the novelty of the work. The experimental validation of the ESGONLP-HSC model is carried out, and the results are examined under diverse aspects. The experimentation outcomes reported the promising performance of the ESGONLP-HSC model over recent state of art approaches.

Keywords: natural language processing; social networking; hate speech; deep learning; data classification

Citation: Asiri, Y.; Halawani, H.T.; Alghamdi, H.M.; Abdalaha Hamza, S.H.; Abdel-Khalek, S.; Mansour, R.F. Enhanced Seagull Optimization with Natural Language Processing Based Hate Speech Detection and Classification. *Appl. Sci.* **2022**, *12*, 8000. https://doi.org/10.3390/app12168000

Academic Editors: Evgeny Nikulchev and Vladimir Borisovich Barakhnin

Received: 27 June 2022
Accepted: 5 August 2022
Published: 10 August 2022

1. Introduction

Social media is altering the face of transmission and culture of communities across the globe. Globally, the quantity of social media users has increased significantly in recent times, in spite of the lower quality of internet facilities and the infrequent disruptions or blocking of mass media sites in the country [1]. Diverse populations in the country were utilizing online mass media for expressing opinions, communicating, sharing information, and chatting with friends. However, the mobility and anonymity of online mass media allow the internet user behind the screen in order to scatter odious content [2]. Mass media platforms, such as Twitter and Facebook, are being condemned because of not preventing hate speech (HS) on their sites, or their face criticism for the fact that their actions against HS were taken under pressure. To prohibit and control HS, governments across the globe are forming rigid rules and continue implementing these kinds of rules using surveillance

within their circumference [3]. Moreover, government agencies have already launched a law that extends the anti-terrorism law to include cyberspace in order to prohibit the dispersion of any terrorizing or coarse information [4,5].

Distinguishing text that comprises HS is not an easy task, even for human beings [6]. Manual judgment of HS is not only time-consuming but also establishes a personal sense of HS composition. Thus, the meaning of HS is obviously important in drafting a rule for the annotation function of the dataset, both for the annotator, and for ensuring the easy functioning of the automated model assessment [7]. Many research studies on mass media have described HS as a language that assaults or directs hate towards groups, depending on particular features namely, gender, race, political views, religious affiliation, physical appearance, ethnic origin, and so on. The definition marks that HS language encourages ferocity or hatred toward groups [8]. There is also an acknowledgment that it is more possible that HS on mass media is associated with real hate offenses. Though there exists another type of speech whose definition is equivalent to HS, it exists on a distinct level [8]. One such instance of such type of speech is offensive speech which is utilized for hurting somebody. The rhetoric disparity or indirect verbal act is a key factor in recognizing something is offensive or HS. Various studies are utilized for estimating HS on mass media websites. Hate content and Semantic text are traced by implying Artificial Intelligence (AI) and natural language processing (NLP) methods.

Automatic HS identification and prediction method should assure that the methodology is sustainable dependable, and expandable, owing to the numerous quantities of Internet content. The automated method suggested in this article interprets text content as non-harmful and HS. Due to the continuous deepening of the DL models, the parameter count involved in the DL models gets increased rapidly and leads to model overfitting. Furthermore, distinct hyperparameters have a considerable influence on the performance of the CNN model. Specifically, the hyperparameters namely number of epochs, batch size, and learning rate selection are required to accomplish enhanced results. As the traditional trial and error method for hyperparameter tuning is a tiresome and erroneous process, metaheuristic algorithms can be applied. Therefore, in this work, we employ a metaheuristic algorithm for the hyperparameter selection of the DL model.

This study focuses on the design of an Enhanced Seagull Optimization with Natural Language Processing Based Hate Speech Detection and Classification (ESGONLP-HSC) model. The ESGONLP-HSC model majorly aims at the recognition and classification of HS on social media. The presented ESGONLP-HSC model involves data pre-processing at several stages, such as tokenization, vectorization, etc. Additionally, the Glove technique is applied for the feature extraction process. Finally, SGO with attention-related bidirectional long short-term memory (ABLSTM) model is utilized for the classification of social media text into three classes such as offensive, hate, and neutral language. The experimental validation of the ESGONLP-HSC model is carried out, and the outcomes are examined under diverse aspects.

The rest of the paper is organized as follows. Section 2 offers a detailed literature review of hate speech classification. Then, Section 3 elaborates on the presented ESGONLP-HSC model and Section 4 validates the performance of the proposed model. Lastly, Section 5 concludes the study, and Section 6 provides practical implications.

2. Related Works

In [9], the authors presented an effectual Convolutional, BiGRU, and Capsule network-based deep learning model, HCovBi-Caps for hate speech classification. The presented model is validated on two Twitter-based benchmark datasets—DS1 (balanced) and DS2 (unbalanced). In [10], the authors developed BiCHAT model to represent tweets for hate speech classification. It receives the tweets as input and passed them to the BERT layer trailed by the convolutional encoded representation which again fed into the attention-aware Bidirectional LSTM network. At last, the model labeled the tweet as hateful or normal using a softmax layer.

The researchers in [11] endeavor to examine the influence of pre-processing on Arabic offensive language categorization. It reviews six pre-processing approaches: normalization of diverse types of Arabic letters, normalization of selected nouns from dialectal Arabic, and translation of emojis into Arabic textual labels. The outcomes validate substantial differences in the impact of pre-processing over every classifier kind and every dataset. Pham et al. [12] suggest a way of adapting the general purpose RoBERTa language system to a particular text categorization task, that is, Vietnamese HS Detection. It first of all tunes the PhoBERT 1 [9] on the dataset through retraining the model over the Masked Language Model (MLM) task and later employing its encoder for the purpose of text categorization. Robinson et al. [13] establish a feature selection (FS) interpretation in their analysis of a task utilizing Twitter as a case study and display results that can challenge classical perceptions of the significance of physical feature engineering: automatic FS may greatly decrease prudently engineered structures above 90% and choose mainly generic features which were frequently employed by most language-oriented missions.

Awal et al. [14] recommend an innovative multitask learning-related system, AngryBERT, that equally studies HS recognition with sentiment categorization and target recognition as subsidiary related tasks. The wide experiments were conducted for augmenting three commonly utilized HS recognition datasets. The researchers in [15] aim to predict its distinct types accurately through the exploration of a collection of text mining structures. Two different groups of features can be examined for issue suitability. All those were considered baseline features and new or self-discovered features. Baseline features comprise the commonly employed efficient features of related studies. Mohtaj et al. [16] provide TU Berlin team experimentations and outcomes on tasks 1B and 1A of the shared task on offensive content and identification HS in Indo-European classics 2021. The victory of distinct Natural Language Processing methods can be assessed for the particular subtasks all over the competition.

Kumar et al. [17] suggest a quaternion neural network (NN)-related method with extra fusion elements for every pair of modalities. The method can be tested on the MMHS150K Twitter datasets for the classification of HS. The algorithm displays a nearly 75% decrement in variables and further facilitates training time and storage space. The researchers in [18] recommend two new feature extraction techniques which utilize the previous sentic computing sources, AffectiveSpace and SenticNet. Such techniques were potential methods for deriving affect-aware depictions from the text. Moreover, this article offers a machine learning (ML) structure with the use of assembling diverse features for improvising the complete classification outcome. Succeeding the description of this technique, it further learns the impacts of known FS methodologies like SIMilarity-related sentiment projectiON (SIMON) and TF-IDF.

The authors in [19] proposed a Bayesian technique using the Monte Carlo dropout method in the attention layer of the transformer model for providing highly calibrated reliability estimation. The authors evaluated and visualized the outcome of the presented model on the hate speech classification problem in different languages. In addition, the authors have tested the ability of the affective dimension to enhance the data derived by the BERT approach to the classification of hate speech. The authors in [20] analyzed the gender bias in various datasets and presented an ensemble learning model depending upon various feature spaces to detect hate speech by learning from distinct problem abstractions such as unintended bias evaluation measures. A set of nine distinct feature spaces are used for training the classification model and validated on open access dataset. Cruz et al. [21] devised a model to inspect the relation among many feature extractors and classifiers for understanding them. It can be utilized for selecting a collection of complementary approaches for composing a robust multiple classifier system (MCS) to detect hate speech.

Yao et al. [22] developed a dynamic statistical model for learning time-aware word vector representation. The presented model can concurrently learn the time-aware embeddings and resolves the resultant "alignment issue". This model undergoes training on a crawled *New York Times* dataset. Additionally, many intuitive evaluation models

are developed for temporal word embeddings. Hong et al. [23] introduced a supervised model to extract relations. The SVM model is applied for the recognition and classification of relations in the Automatic Content Extraction (ACE) corpus. A collection of features comprising lexical tokens, syntactic structures, and semantic entity kinds relation detection and classification process is presented. Zhang et al. [24] developed a novel approach by the use of a deep neural network and integration of convolutional and gated recurrent networks. A widespread experimental analysis is carried out on the open access Twitter dataset. Kim et al. [25] defined an easier neural language model which is based on the character level input. The prediction process is carried out at the word level. It uses a CNN model and a highway network over characters, where the outcome is passed to a long short-term memory (LSTM) recurrent neural network language model (RNN-LM).

Numerous automated tools have been available in the literature for effective hate speech detection and classification. In spite of the ML and DL models that existed in the earlier studies, it is still needed to enhance the hate speech classification performance. To resolve the difficult and erroneous hyperparameter tuning process, metaheuristic algorithms can be applied. Therefore, in this work, we employ the EGSO algorithm for the hyperparameter selection of the ABLSTM model.

3. The Proposed ESGONLP-HSC Model

In this study, a new ESGONLP-HSC model has been introduced to identify and classify the occurrence of HS on social media websites. Primarily, the presented ESGONLP-HSC model involves data pre-processing at several stages, such as tokenization, vectorization, etc. Additionally, the Glove technique is applied for the feature extraction process. Next, the ESGO-ABLSTM model is utilized for the classification of social media text. Figure 1 depicts the block diagram of the ESGONLP-HSC approach. The proposed model involves different steps as listed in the following:

- Data Preprocessing
- Feature Extraction
- Data Classification
- Hyperparameter Tuning

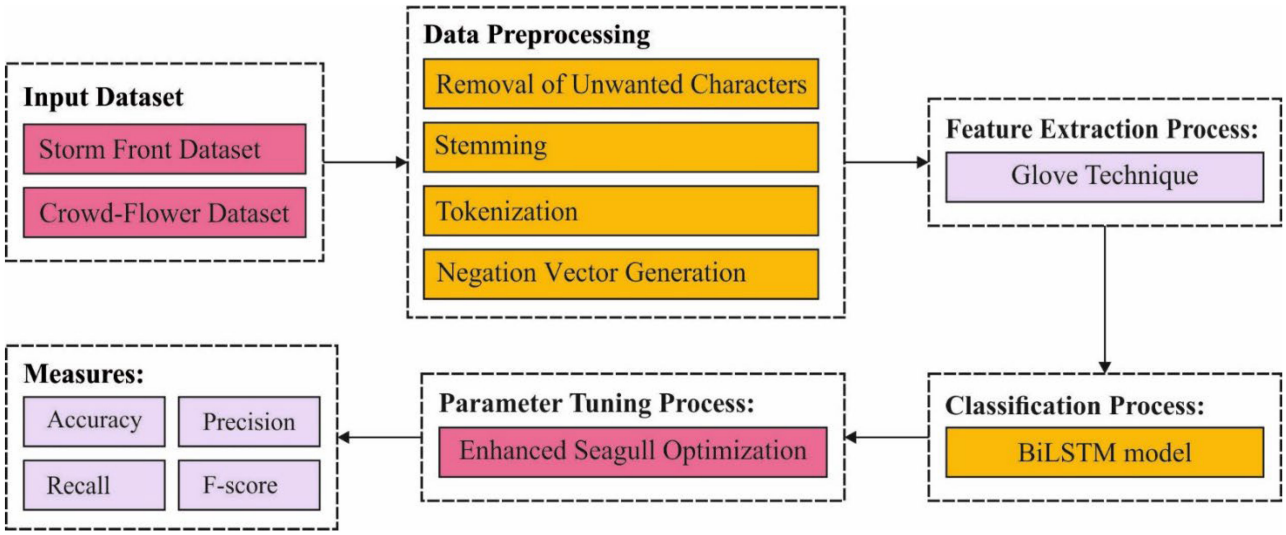

Figure 1. Block diagram of ESGONLP-HSC technique.

3.1. Data Pre-Processing

The Twitter data gathered in this method shall always be extraneous information; this raises the complication of recognizing HS on Twitter. Therefore, NLP has performed numerous key tasks for enhancing the procedure of recognizing HS:

- Unnecessary characters such as '\', '/', '#', '@', etc. were eliminated from Twitter data.
- Once unwanted characters are removed, the stemming process takes place, which derives the root words in the tweet. It is an effective NLP model due to the fact that it effectively processes words by the identification of the root or origin. At this stage, a lookup table is used for managing the incoming words related root words. It determines origin words using a suffix stripping process.
- NLP tokenization was the following step, where tweets were examined with the help of the OpenNLP tool. It uses sentDetect() function for identifying the start and end boundaries of all sentences. Once each sentence is recognized, the tokenization splits the sentence into smaller sentences.
- At last, a negation vector was produced with the help of the lookup table. The table encompasses root words and mixed stemmed words. Then, negative words are examined and a few of them are allocated a -1 value. The rest of the words are allocated a +1 value.

3.2. Glove-Based Feature Extraction

Once the social media data are pre-processed, the Glove technique is applied for the feature extraction process. Pennington et al. [26] proposed a Glove that signifies "Global Vector" as an alternative word embedding model. Word2vec methodology learns semantics of words bypassing a local content window through the training dataset line-by-line for predicting a word from the surroundings or surroundings of a presented word. The study discusses that the local content window model is sufficient for extracting semantics amongst words and does not exploit the count-based statistical dataset with respect to word co-occurrence. Local content window and count-based matrix factorization methodologies are consolidated in Glove to obtain a better description. Glove make use of matrix factorization for getting an accumulative global co-occurrence statistic of word-word from the information.

3.3. Hate Speech Classification

In this work, the ABLSTM model is utilized for the classification of social media text into three classes such as neutral, offensive, and hateful language. The LSTM approach is elected because the data are suitable and preserved and those data are removed depending on the dataset it trains with. The LSTM is extremely utilized in several NLP tasks such as document classification, sentiment analysis, etc. [27]. The LSTM cell utilized in this article is an input, output, and forget layer. According to the figure, the LSTM cell mathematical formulated as follows:

$$f_z = \sigma\left(W_{fh}h_{z-1} + W_{fx}x_z + b_f\right) \quad (1)$$

$$i_z = \sigma(W_{ih}h_{z-1} + W_{ix}x_z + b_i) \quad (2)$$

$$\widetilde{c}_z = \tan\mathrm{h}(W_{\widetilde{c}h}h_{z-1} + W_{\widetilde{c}x}x_z + b_{\widetilde{c}}) \quad (3)$$

$$c_z = f_z \,.\, c_{z-1} + i_z \,.\widetilde{c}_t \quad (4)$$

$$o_z = \sigma(W_{oh}h_{z-1} + W_{ox}x_z + b_0) \quad (5)$$

$$h_z = 0_z \cdot \tan\mathrm{h}(c_z) \quad (6)$$

whereas x_z implies the input; h_{z-1}, and h_z signifies the outcome of the previous LSTM unit and present outcome; c_{z-1}, and c_z implies the memory in the final LSTM unit and cell state; f_z stands for the forget gate value; W_i, $W_{\widetilde{c}}$, and W_0 stands for the weighted; b represents the bias; the operator $'.'$ defines the pointwise multiplication of 2 vectors. During the LSTM, the input gate is decided and novel data are stored from the cell state, the resultant gate is also decided; data are outcome-dependent upon the cell state. By integrating the concepts of BRNN and LSTM, it can be feasible for achieving Bi-directional LSTM (BiLSTM) that is superior and more efficient than LSTM from classifier procedures, particularly from data classification tasks. Figure 2 showcases the framework of the BiLSTM approach.

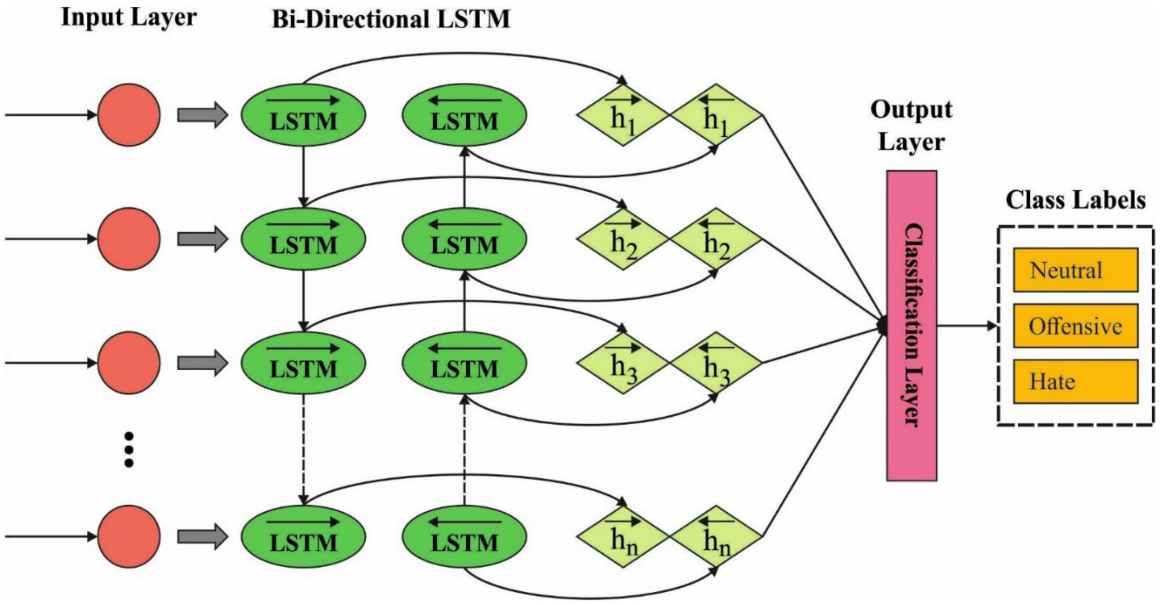

Figure 2. Structure of BiLSTM.

In the ABLSTM model, attention methodologies were utilized for assigning diverse weights to terms contributing diversely to the sentiment of a text. The general means of assigning distinct weights to diverse terms in a sentence is using a weighted combination of every hidden state, S_{A_W} as follows.

$$\alpha_t = \frac{\Sigma xp(v^T \cdot \widetilde{h})}{\sum\limits_t \exp\,(v \cdot \widetilde{h})} \tag{7}$$

$$S_{A_w} = \sum_t \alpha_t h_t \tag{8}$$

whereas $\widetilde{h}$ and h are well-defined and v is a trainable variable.

3.4. Hyperparameter Optimization

In this work, the ESGO algorithm is utilized as a hyperparameter optimizer to adjust the hyperparameters related to the ABLSTM model. The SGO is a novel SI metaheuristics algorithm, and it is inspired by the behavior of seagulls in their colonies, especially by their attacking (hunting) and migration strategies [28]. On one hand, they attack other birds over the sea during migration. On the other hand, they create spiral shape movements to attack the prey efficiently. SGO algorithm is formulated by considering these behaviors. It is noteworthy that SGO shows better performance on global bound-constrained optimization problems. Thus, we assume that it performs well in many real-time problems. Nevertheless, the basic SGO attains considerable outcomes for unconstrained benchmarks; by conducting further research on the CEC test suite, it has been noted that the convergence speed can be optimized.

Even though the original SGO managed to find the optimum searching region in every run, in some runs, it does not converge satisfactorily, and consequently, the last generated result suffers from poor quality. In order to address these shortcomings, the opposition-based learning (OBL) process has been devised. Based on the preceding analysis, it has been proved that the OBL algorithm will considerably increase diversification and intensification. Following every iteration, once the optimal solution is defined X_{best}, their opposite solution X^o_{best} is generated by the subsequent formula for each variable j:

$$X^o_{best,j} = l_j + u_j - X_{best,j} \tag{9}$$

In Equation (9), $X^{o}_{best,j}$ indicates opposite $j-$th dimension optimal solution, l_j and u_j denotes lower and upper limits of $j-$th parameter, correspondingly. Based on the fitness function, a greedy selection is used between the opposite and initial optimal solution, and the best one is preserved for the following iteration. The suggested methodology is called enhanced SGO (ESGO), and their pseudocode has been shown in Algorithm 1. At last, from the perception of computation difficulty, ESGO contributes slightly towards the total difficulty. More accurately, it adds an additional calculation to all the iterations. That computation is an OBL process that selects between X_{best} and X^{o}_{best} through a greedy selection. In another word, $\mathcal{O}((N+(N+1)) * \mathrm{Max}_{iterations})$, while N signifies individual count where $\mathrm{Max}_{iterations}$ represent the iteration count.

The ESGO approach extracts a fitness function for attaining enhanced classification outcomes. It fixes a positive integer for indicating the better execution of the candidate solutions. In this article, the reduction of the classification error rate is treated as a fitness function, as given in Equation (10).

$$fitness(x_i) = ClassifierErrorRate(x_i) = \frac{number\ of\ misclassified\ samples}{Total\ number\ of\ samples} * 100 \quad (10)$$

Algorithm 1: Pseudocode of ESGO Algorithm

Input: Seagull population $\boldsymbol{P_s}$
Output: Optimum searching agent $\boldsymbol{P_{bs}}$
Parameter initialization: $\boldsymbol{A}$, $\boldsymbol{B}$ and $\mathbf{Max}_{\boldsymbol{itteration}}$
Assume $\boldsymbol{f_c} \leftarrow \mathbf{2}$
Assume $\boldsymbol{u} \leftarrow \mathbf{1}$
Assume $\boldsymbol{v} \leftarrow \mathbf{1}$
while $\boldsymbol{x} < \mathbf{Max}_{\boldsymbol{iteration}}\boldsymbol{s}$ do
/*determine fitness values of seagulls$\boldsymbol{l}^*$/
for $\boldsymbol{i} = \mathbf{1}$ to $\boldsymbol{n}$ (every dimension), do
$\boldsymbol{FIT_s}[\boldsymbol{i}] \leftarrow$ Fitness_Function $(\boldsymbol{P_s}(\boldsymbol{i},:))$
end for
/*elect optimal fitness value*/
$\boldsymbol{Best} = \boldsymbol{FIT_s}[0]$
for $\boldsymbol{i} = \mathbf{1}$ to $\boldsymbol{n}$ do
if $\boldsymbol{FIT_s}[\boldsymbol{i}] < \boldsymbol{Best}$ then
$\boldsymbol{Best} \leftarrow \boldsymbol{FIT_s}[\boldsymbol{i}]$
end if
end for
/*elect fitness value for searching agent*/
$\boldsymbol{P_{bs}} = \boldsymbol{Best}$
$/ * \boldsymbol{Migration}^* /$
$\boldsymbol{rd} \leftarrow \boldsymbol{Rand}(\mathbf{0},\mathbf{1})$
$\boldsymbol{k} \leftarrow \boldsymbol{Rand}(\mathbf{0},\mathbf{2\pi})$
$/^*\boldsymbol{Attacking}^* /$
$\boldsymbol{r} = \boldsymbol{u} \times \boldsymbol{e}^{\boldsymbol{kv}}$
$\boldsymbol{D_s} = |\boldsymbol{C_s} + \boldsymbol{M_s}|$
$\boldsymbol{P} \leftarrow \boldsymbol{x'} \times \boldsymbol{y'} \times \boldsymbol{z'}$
$\boldsymbol{P_s}(\boldsymbol{x}) = (\boldsymbol{D_s} \times \boldsymbol{P}) + \vec{\boldsymbol{P_{bs}}}(\boldsymbol{x})$
$\boldsymbol{x} \leftarrow +\mathbf{1}$
Carry out OBL process
Elect $\boldsymbol{X_{best}}$ and $\boldsymbol{X^o_{best}}$ via greedy selection
end while
return $\boldsymbol{P_{bs}}$

4. Results and Discussion

The ESGONLP-HSC model is simulated using Python 3.6.5 tool on a PC i5-8600k, GeForce 1050Ti 4GB, 16GB RAM, 250GB SSD, and 1TB HDD. The hyperparameter values are given in Table 1.

Table 1. Hyperparameter Setting.

Hyperparameter	Value
Learning rate	0.01
Dropout	0.5
Batch size	5
Number of epochs	50
Activation function	ReLU

4.1. Dataset Details

The performance validation of the ESGONLP-HSC model is tested using two datasets namely Storm front and Crowd-flower datasets. The first Stormfront dataset [29] holds 10,568 sentences. A total number of 10,568 sentences have been extracted from Stormfront and categorized into two class labels namely hate and no-hate. Furthermore, each sentence holds additional information such as a post identifier and the sentence's position in the post. After preprocessing, a total of 1119 samples are grouped into the hateful class and 8537 samples fall into the not hateful class. This information makes it possible re-build the conversations these sentences belong to. Next, the CrowdFlower [30] comprises 24,783 tweets which 1430 hate tweets, 19,190 offensive (hate) tweets, and 4163 normal tweets. In this work, binary classification is performed where 4163 tweets fall into normal class and the rest of the samples come under hate class. For experimental validation, a ten-fold cross-validation technique is used.

4.2. Result Analysis

Table 2 provides a detailed comparative accuracy analysis of the ESGONLP-HSC model with existing methods such as KNLPE-DNN [5], TWE Model [22], SVM [23], CG-DNN [24], and CANL-NN [25]. Figure 3 inspects a comparative $accu_y$ results of the ESGONLP-HSC model on the test Storm front dataset. The figure implied that the ESGONLP-HSC model has resulted in enhanced $accu_y$ values over other models. For instance, with 500 tweets, the ESGONLP-HSC model has gained an increased $accu_y$ of 99.24% whereas the KNLPE-DNN, TWE, SVM, CG-DNN, and CANL-NN models have attained a reduced $accu_y$ of 98.92%, 98.29%, 97.88%, 96.81%, and 95.50%, respectively. Additionally, with 2500 tweets, the ESGONLP-HSC system has obtained an increased $accu_y$ of 99.17% whereas the KNLPE-DNN, TWE, SVM, CG-DNN, and CANL-NN techniques have attained a reduced $accu_y$ of 98.75%, 98.14%, 97.85%, 96.31%, and 95.92%, correspondingly.

Figure 4 examines a comparative $accu_y$ outcomes of the ESGONLP-HSC system on the test Crowd-flower dataset. The figure implied that the ESGONLP-HSC technique has resulted in enhanced $accu_y$ values over other models. For instance, with 500 tweets, the ESGONLP-HSC approach has attained an increased of 99.22% whereas the KNLPE-DNN, TWE, SVM, CG-DNN, and CANL-NN approaches have acquired a reduced $accu_y$ of 98.71%, 98.11%, 97.65%, 97.50%, and 96.46%, correspondingly. Moreover, with 2500 tweets, the ESGONLP-HSC methodology has obtained an increased of 99.12% whereas the KNLPE-DNN, TWE, SVM, CG-DNN, and CANL-NN systems have attained a reduced accu_y of 98.31%, 98.07%, 97.66%, 96.48%, and 95.54% correspondingly.

Table 2. Accuracy analysis of ESGONLP-HSC method with the existing approach under Storm front and Crowd-flower datasets.

Accuracy (%)						
No. of Tweets	**ESGONLP-HSC**	**KNLPE-DNN**	**TWE Model**	**SVM**	**CG-DNN**	**CANL-NN**
Storm front Dataset						
500	99.24	98.92	98.29	97.88	96.81	95.50
1000	98.95	98.48	98.29	97.54	96.78	95.87
1500	98.97	98.41	98.23	97.55	96.68	96.00
2000	99.01	98.43	98.11	97.38	97.43	95.67
2500	99.17	98.75	98.14	97.85	96.31	95.92
Crowd-flower Dataset						
500	99.22	98.71	98.11	97.65	97.50	96.46
1000	99.04	98.35	98.22	97.83	96.38	95.77
1500	99.08	98.55	98.18	97.34	97.00	94.76
2000	99.18	98.65	98.04	97.42	96.70	95.86
2500	99.12	98.31	98.07	97.66	96.48	95.54

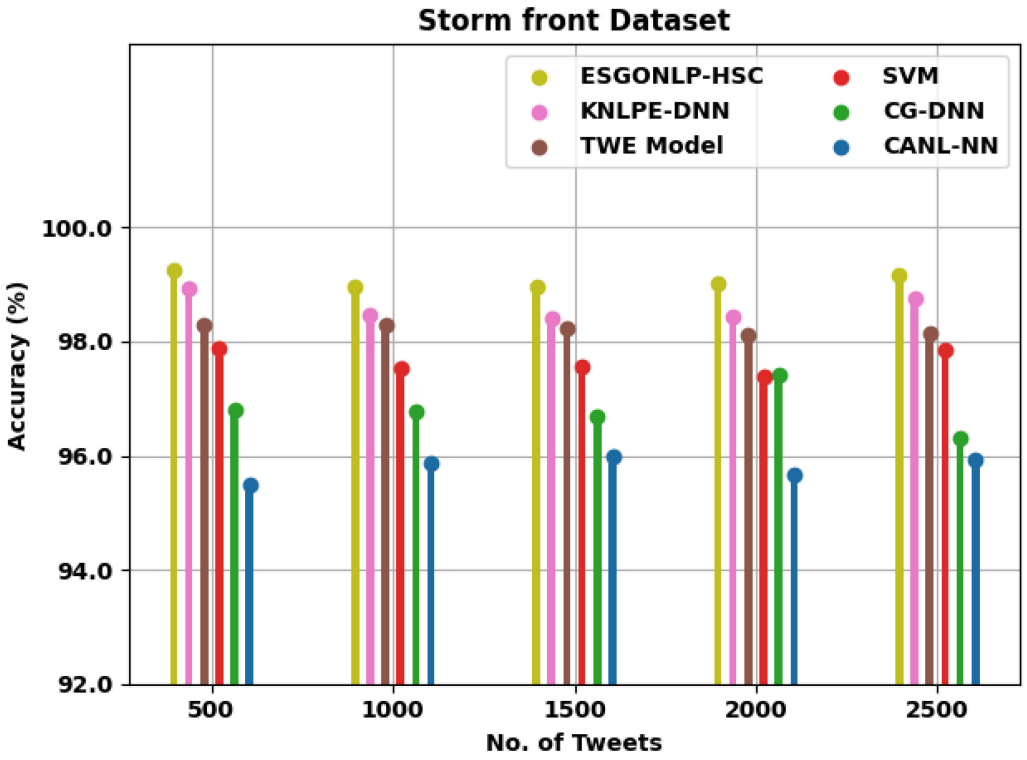

Figure 3. $Accu_y$ analysis of ESGONLP-HSC technique under Storm front dataset.

Table 3 offers a brief relative precision examination of the ESGONLP-HSC technique on the Stormfront and Crowd-flower datasets under different numbers of tweets. Figure 5 examines a comparative $prec_n$ results of the ESGONLP-HSC approach on the test Storm front dataset. The figure implied that the ESGONLP-HSC system has resulted in enhanced $prec_n$ values over other models. For example, with 500 tweets, the ESGONLP-HSC methodology has reached an increased $prec_n$ of 99.26% whereas the KNLPE-DNN, TWE, SVM, CG-DNN, and CANL-NN algorithms have obtained a reduced $prec_n$ of 98.32%, 98.22%, 97.67%, 96.69%, and 95.82%, correspondingly. Furthermore, with 2500 tweets, the ESGONLP-HSC methodology has acquired an increased $prec_n$ of 99.20% whereas the KNLPE-DNN, TWE,

SVM, CG-DNN, and CANL-NN methodologies have reached a reduced $prec_n$ of 98.43%, 98.26%, 97.73%, 96.92%, and 94.66%, correspondingly.

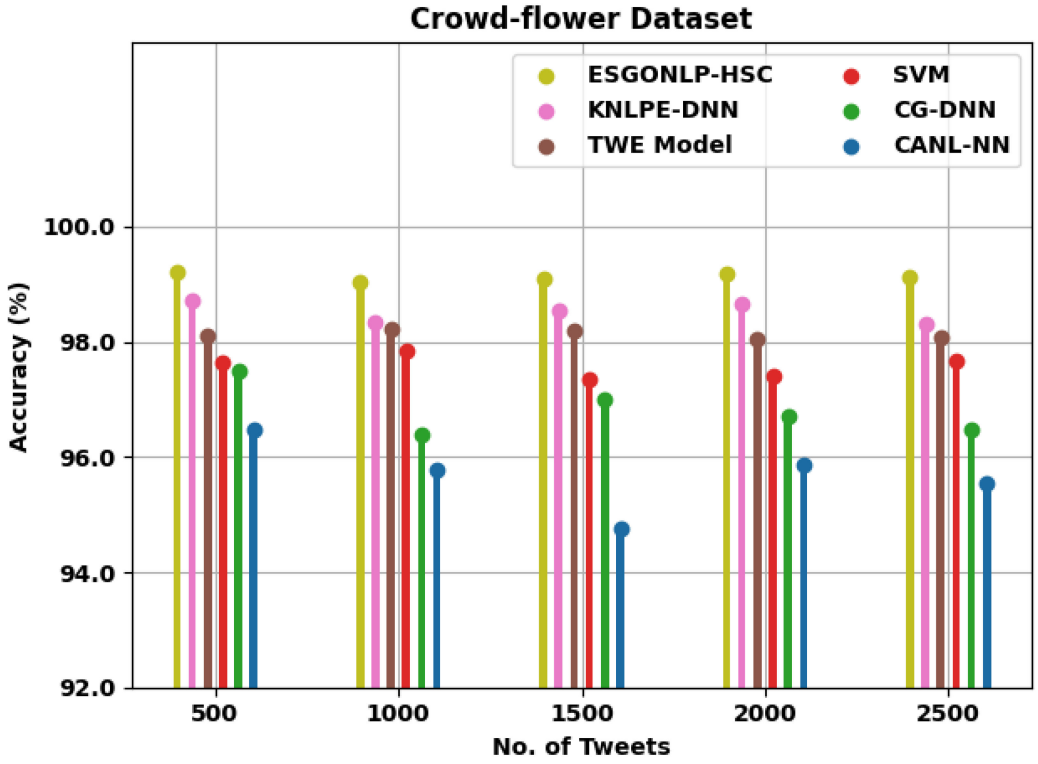

Figure 4. $Accu_y$ analysis of ESGONLP-HSC technique under Crowd-flower dataset.

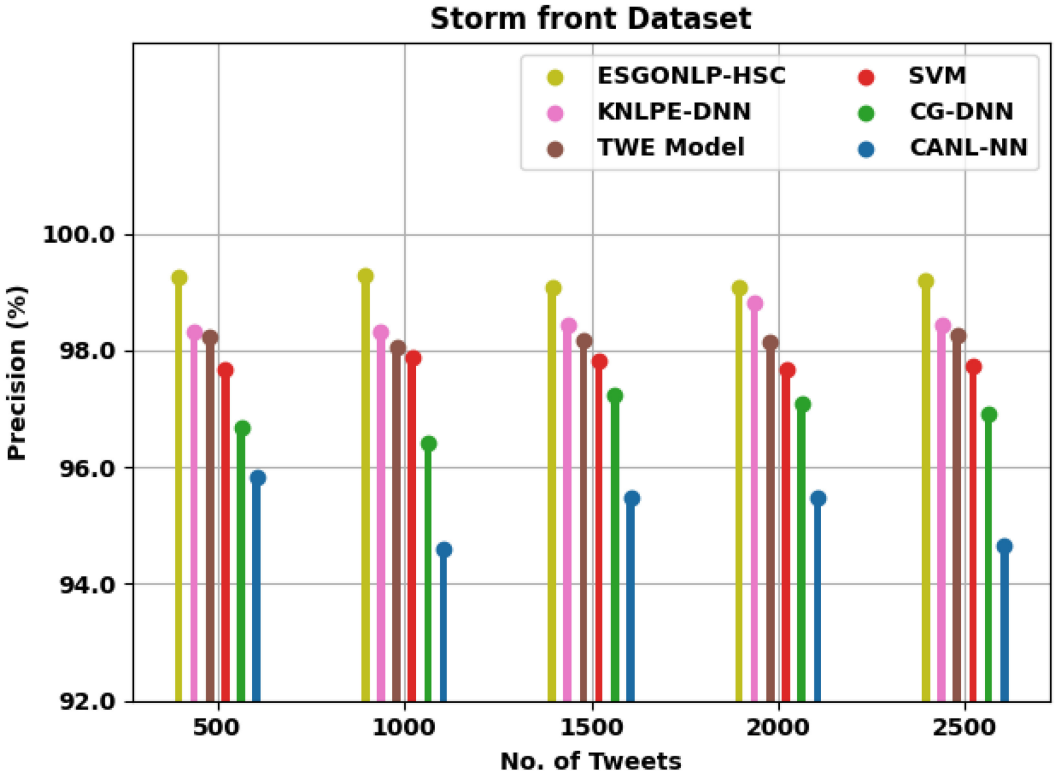

Figure 5. $Prec_n$ analysis of ESGONLP-HSC technique under Storm front dataset.

Table 3. Precision analysis of ESGONLP-HSC method with existing approach under Storm front and Crowd-flower datasets.

Precision (%)						
No. of Tweets	**ESGONLP-HSC**	**KNLPE-DNN**	**TWE Model**	**SVM**	**CG-DNN**	**CANL-NN**
Storm front Dataset						
500	99.26	98.32	98.22	97.67	96.69	95.82
1000	99.29	98.31	98.06	97.89	96.41	94.59
1500	99.07	98.45	98.16	97.82	97.25	95.47
2000	99.07	98.82	98.15	97.69	97.08	95.49
2500	99.20	98.43	98.26	97.73	96.92	94.66
Crowd-flower Dataset						
500	99.22	98.80	98.11	97.38	96.78	95.09
1000	99.02	98.77	98.10	97.46	97.37	96.05
1500	99.27	98.90	97.98	97.88	96.56	95.76
2000	99.15	98.46	98.02	97.86	97.04	96.43
2500	99.00	98.82	98.23	97.83	97.44	94.55

Figure 6 scrutinizes the comparative $prec_n$ outcomes of the ESGONLP-HSC system on the test Crowd-flower dataset. The figure implied that the ESGONLP-HSC technique has resulted in enhanced $prec_n$ values than other models. For example, with 500 tweets, the ESGONLP-HSC approach has obtained an increased $prec_n$ of 99.22% whereas the KNLPE-DNN, TWE, SVM, CG-DNN, and CANL-NN approaches have gained a reduced $prec_n$ of 98.80%, 98.11%, 97.38%, 95.78%, and 95.09%, correspondingly. In addition to this, with 2500 tweets, the ESGONLP-HSC methodology has achieved an increased $prec_n$ of 99% whereas the KNLPE-DNN, TWE, SVM, CG-DNN, and CANL-NN methodologies have attained a reduced $prec_n$ of 98.82%, 98.23%, 97.83%, 97.44%, and 94.55%, correspondingly.

Table 4 presents a brief comparative recall examination of the ESGONLP-HSC system on the Stormfront and Crowd-flower datasets under different quantities of tweets. Figure 7 reviews the comparative $reca_l$ results of the ESGONLP-HSC techniques on the test Storm front dataset. The figure implied that the ESGONLP-HSC approach has resulted in enhanced $reca_l$ values over other models. For example, with 500 tweets, the ESGONLP-HSC approach has attained increased $reca_l$ of 98.96% whereas the KNLPE-DNN, TWE, SVM, CG-DNN, and CANL-NN techniques have gained a reduced $reca_l$ of 98.63%, 98.14%, 97.76%, 97.33%, and 94.84%, respectively. Additionally, with 2500 tweets, the ESGONLP-HSC methodology has achieved an increased $reca_l$ of 98.98% whereas the KNLPE-DNN, TWE, SVM, CG-DNN, and CANL-NN methods have obtained a reduced $reca_l$ of 98.64%, 98.14%, 97.80%, 97.27%, and 94.62%, correspondingly.

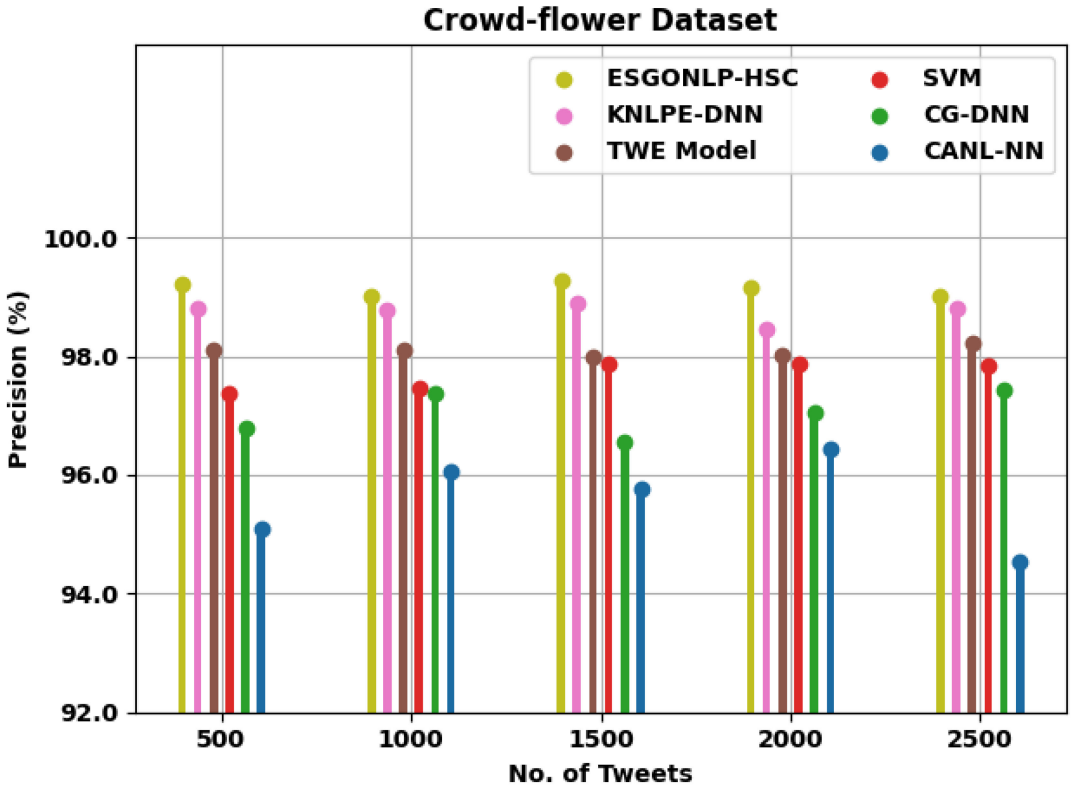

Figure 6. $Prec_n$ analysis of ESGONLP-HSC technique under Crowd-flower dataset.

Table 4. Recall analysis of ESGONLP-HSC method with existing approach under Storm front and Crowd-flower datasets.

Recall (%)						
No. of Tweets	**ESGONLP-HSC**	**KNLPE-DNN**	**TWE Model**	**SVM**	**CG-DNN**	**CANL-NN**
Storm front Dataset						
500	98.96	98.63	98.14	97.76	97.33	94.84
1000	99.20	98.79	97.94	97.44	97.27	94.40
1500	99.26	98.63	98.18	97.37	97.09	94.78
2000	99.16	98.86	97.91	97.85	96.67	96.50
2500	98.98	98.64	98.14	97.80	97.27	94.62
Crowd-flower Dataset						
500	99.23	98.32	98.09	97.36	97.29	95.71
1000	99.33	98.55	98.28	97.33	97.09	96.35
1500	99.04	98.88	98.05	97.65	96.38	94.64
2000	98.98	98.34	98.08	97.60	96.85	94.83
2500	99.00	98.66	98.22	97.88	96.84	94.49

Figure 8 examines the comparative $reca_l$ results of the ESGONLP-HSC system on the test Crowd-flower dataset. The figure implied that the ESGONLP-HSC method has resulted in enhanced $reca_l$ values over other models. For instance, with 500 tweets, the ESGONLP-HSC approach has obtained increased $reca_l$ of 99.23% whereas the KNLPE-DNN, TWE, SVM, CG-DNN, and CANL-NN methodologies have acquired a reduced $reca_l$ of 98.32%, 98.09%, 97.36%, 97.29%, and 95.71%, correspondingly. Along with that,

with 2500 tweets, the ESGONLP-HSC methodology has attained an increased $reca_l$ of 99% whereas the KNLPE-DNN, TWE, SVM, CG-DNN, and CANL-NN approaches have reached a reduced $reca_l$ of 98.66%, 98.22%, 97.88%, 96.84%, and 94.49%, correspondingly.

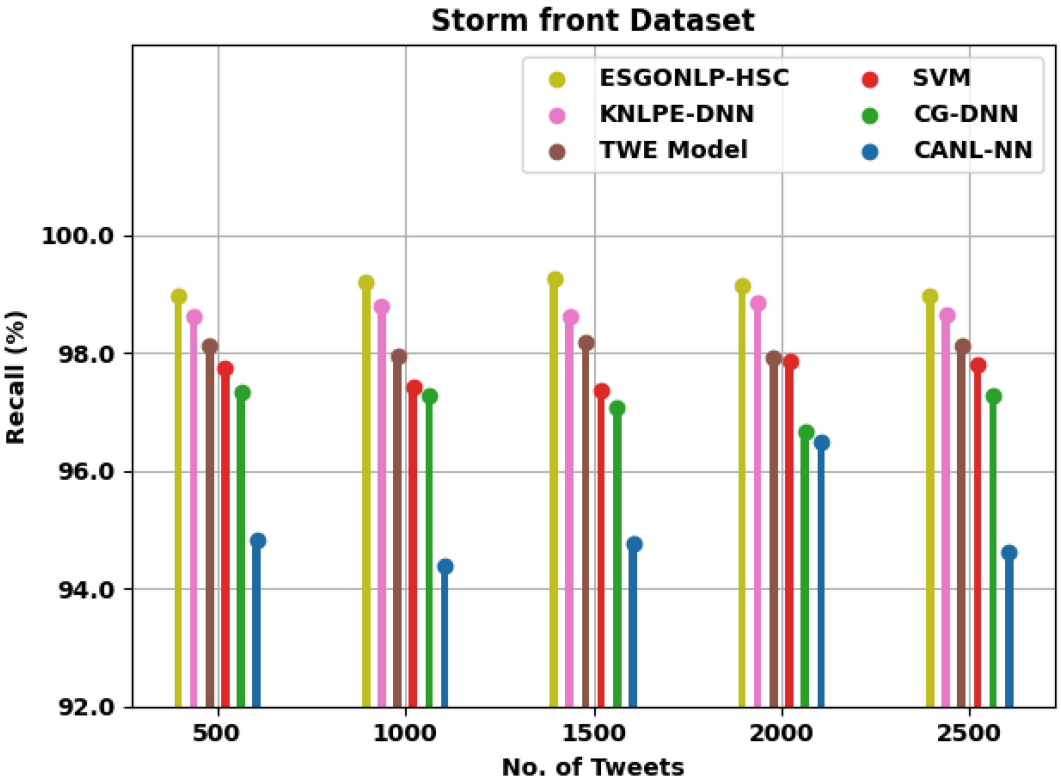

Figure 7. $Reca_l$ analysis of ESGONLP-HSC technique under Storm front dataset.

Table 5 grants a brief comparative F-score analysis of the ESGONLP-HSC technique on the Stormfront and Crowd-flower dataset in different numbers of tweets. Figure 9 evaluates the comparative F_{score} results of the ESGONLP-HSC system on the test Storm front dataset. The figure implied that the ESGONLP-HSC method has resulted in enhanced F_{score} values over other models. For instance, with 500 tweets, the ESGONLP-HSC approach has achieved increased F_{score} of 98.97% whereas the KNLPE-DNN, TWE, SVM, CG-DNN, and CANL-NN algorithms have acquired a reduced F_{score} of 98.60%, 98.24%, 97.47%, 97.36%, and 94.36%, respectively. Moreover, with 2500 tweets, the ESGONLP-HSC system has attained increased F_{score} of 99.02% whereas the KNLPE-DNN, TWE, SVM, CG-DNN, and CANL-NN methodologies have gained a reduced F_{score} of 98.53%, 97.92%, 97.56%, 97.11%, and 95.47%, correspondingly.

Figure 10 assesses the comparative F_{score} outcomes of the ESGONLP-HSC method on the test Crowd-flower dataset. The figure implied that the ESGONLP-HSC system has resulted in enhanced F_{score} values over other models. For example, with 500 tweets, the ESGONLP-HSC methodology has attained increased F_{score} of 99.27% whereas the KNLPE-DNN, TWE, SVM, CG-DNN, and CANL-NN techniques have obtained reduced F_{score} of 98.58%, 98.03%, 97.69%, 96.41%, and 95.78%, correspondingly. Along with that, with 2500 tweets, the ESGONLP-HSC system has obtained increased F_{score} of 99.06% whereas the KNLPE-DNN, TWE, SVM, CG-DNN, and CANL-NN methodologies have obtained a reduced F_{score} of 98.53%, 97.92%, 97.40%, 97.09%, and 95.93%, correspondingly.

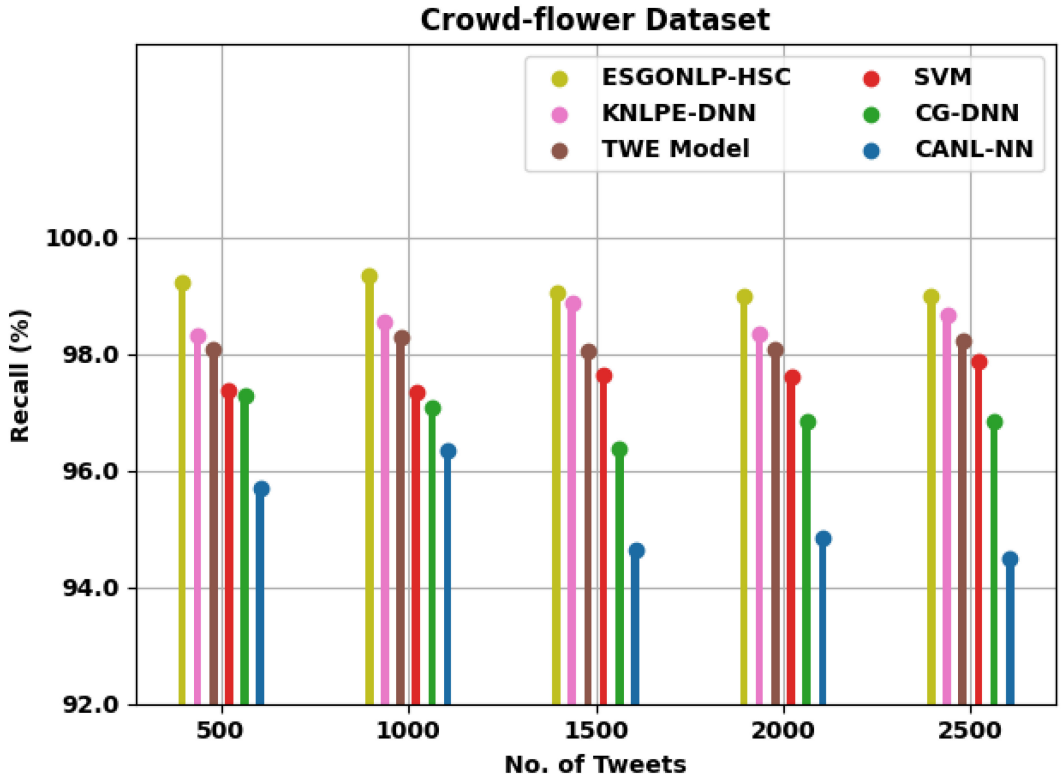

Figure 8. $Reca_l$ analysis of ESGONLP-HSC technique under Crowd-flower dataset.

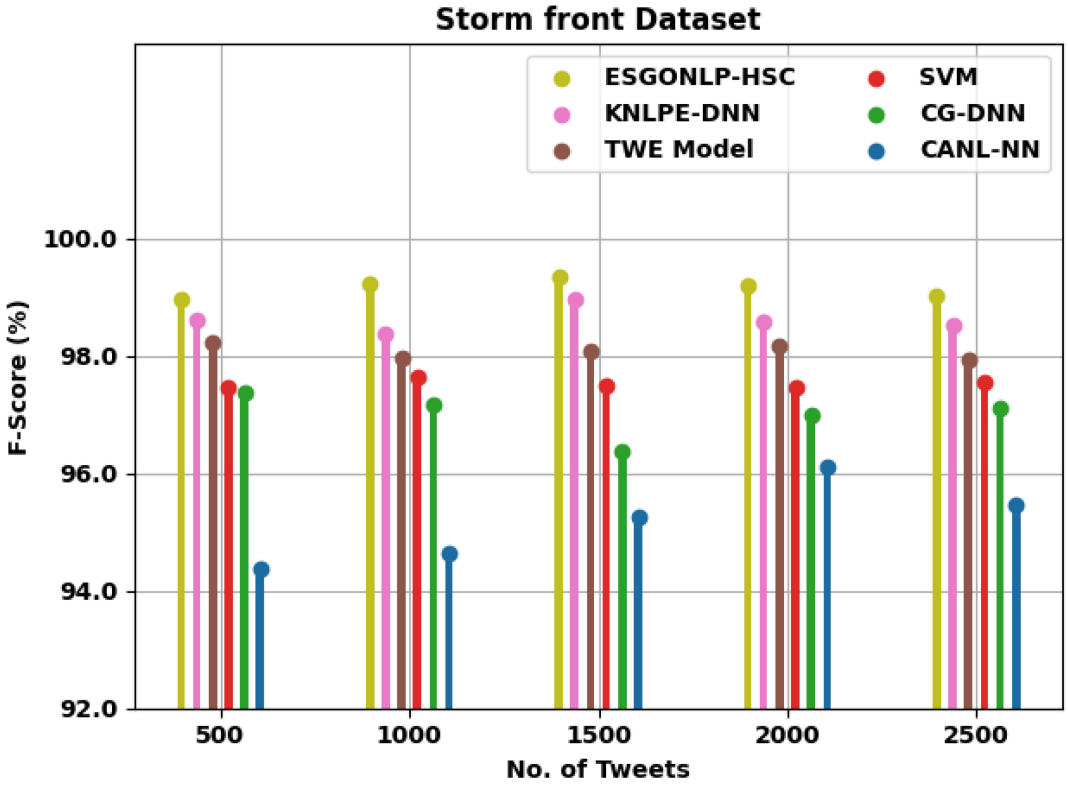

Figure 9. F_{score} analysis of ESGONLP-HSC technique under the Storm front dataset.

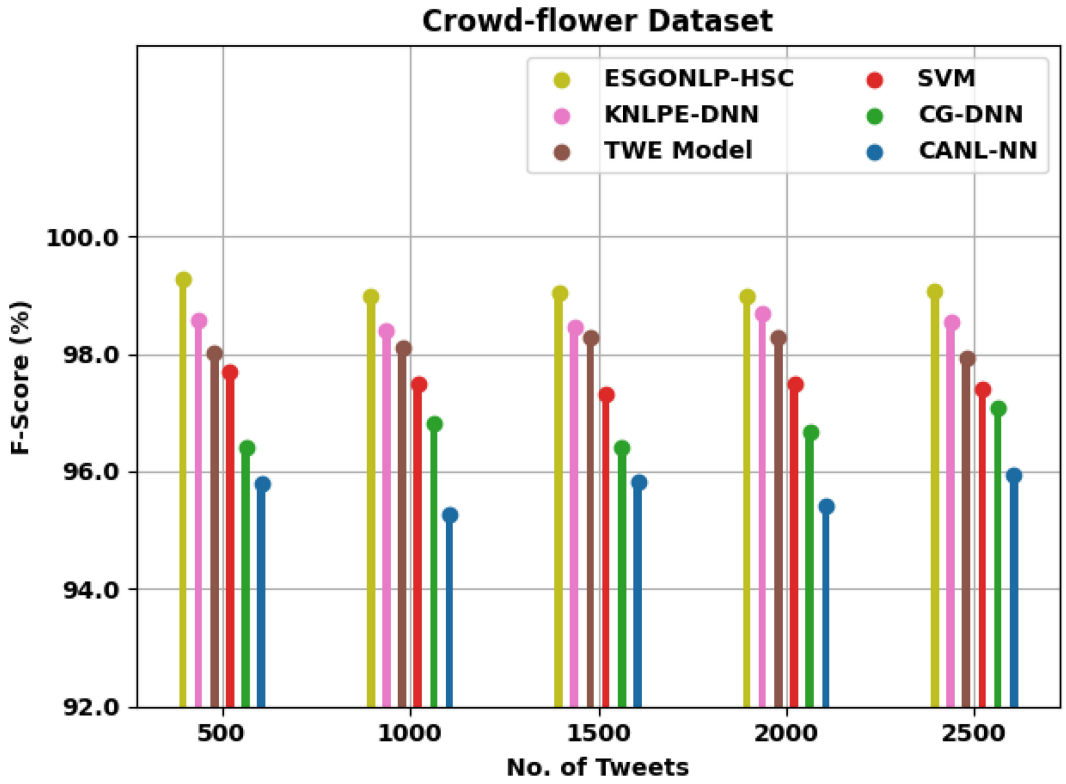

Figure 10. F_{score} analysis of ESGONLP-HSC technique under Crowd-flower dataset.

Table 5. F-score analysis of ESGONLP-HSC method with existing approach under Storm front and Crowd-flower datasets.

F-Score (%)						
No. of Tweets	**ESGONLP-HSC**	**KNLPE-DNN**	**TWE Model**	**SVM**	**CG-DNN**	**CANL-NN**
Storm front Dataset						
500	98.97	98.60	98.24	97.47	97.36	94.36
1000	99.24	98.36	97.96	97.65	97.18	94.64
1500	99.33	98.95	98.07	97.50	96.38	95.26
2000	99.19	98.58	98.18	97.45	96.99	96.11
2500	99.02	98.53	97.92	97.56	97.11	95.47
Crowd-flower Dataset						
500	99.27	98.58	98.03	97.69	96.41	95.78
1000	98.98	98.41	98.09	97.50	96.81	95.26
1500	99.04	98.45	98.29	97.32	96.42	95.81
2000	98.97	98.68	98.27	97.50	96.66	95.42
2500	99.06	98.53	97.92	97.40	97.09	95.93

The training accuracy (TA) and validation accuracy (VA) acquired by the ESGONLP-HSC method on the Storm front dataset is illustrated in Figure 11. The experimental outcome implied that the ESGONLP-HSC methodology has attained maximum values of TA and VA. Specifically, the VA seemed to be higher than TA.

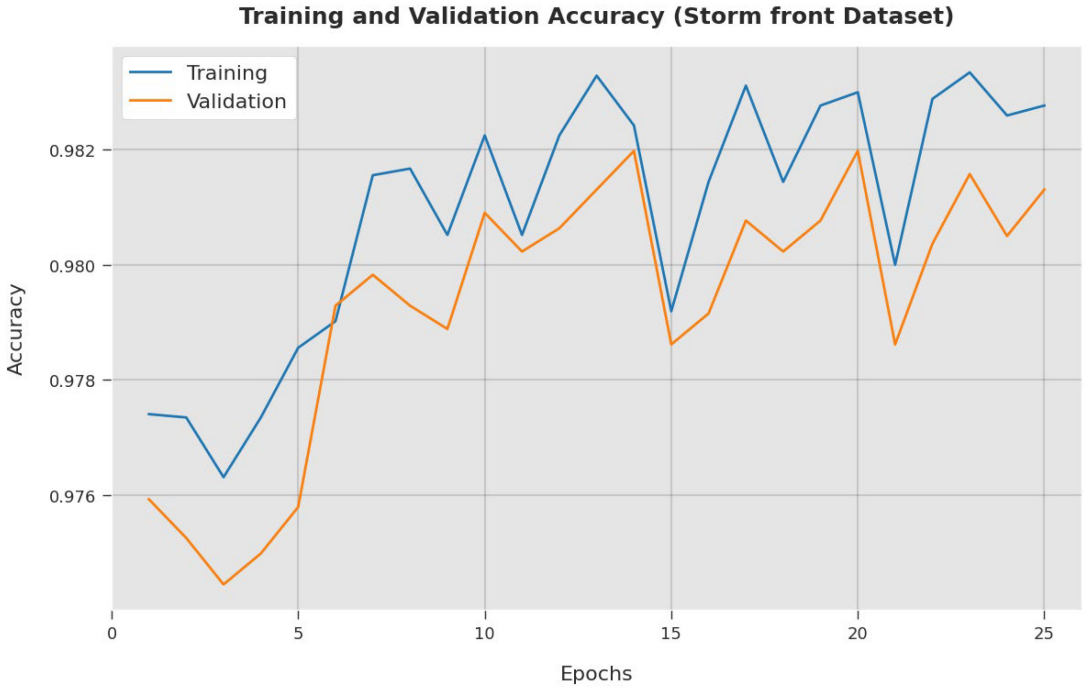

Figure 11. TA and VA analysis of ESGONLP-HSC algorithm under the Storm front dataset.

The training loss (TL) and validation loss (VL) achieved by the ESGONLP-HSC algorithm on the Storm front dataset are established in Figure 12. The experimental outcome inferred that the ESGONLP-HSC technique has accomplished the lowest values of TL and VL. In particular, the VL seemed to be lower than TL.

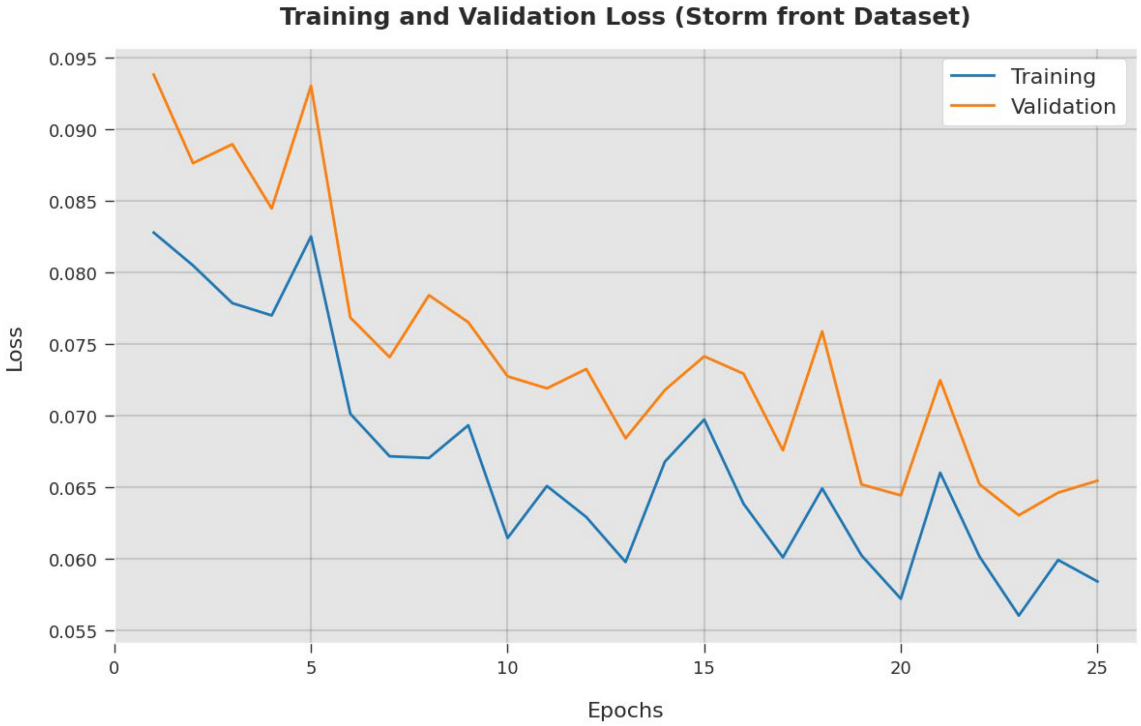

Figure 12. TL and VL analysis of ESGONLP-HSC algorithm under Storm front dataset.

The TA and VA attained by the ESGONLP-HSC system on the Crowd-flower dataset are demonstrated in Figure 13. The experimental outcome implied that the ESGONLP-HSC

technique has gained maximal values of TA and VA. Specifically, the VA seemed to be higher than TA.

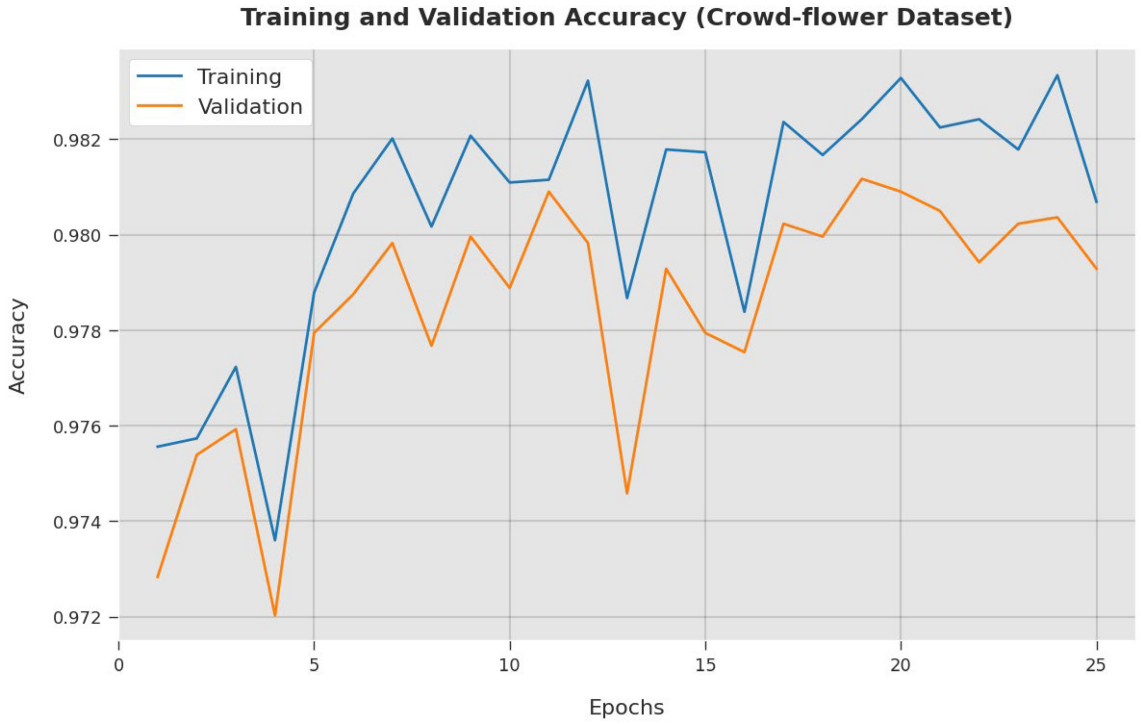

Figure 13. TA and VA analysis of ESGONLP-HSC algorithm under Crowd-flower dataset.

The TL and VL achieved by the ESGONLP-HSC model on Crowd-flower dataset are established in Figure 14. The experimental outcome inferred that the ESGONLP-HSC methodology has accomplished the lowest values of TL and VL. In specific, the VL seemed to be lower than TL.

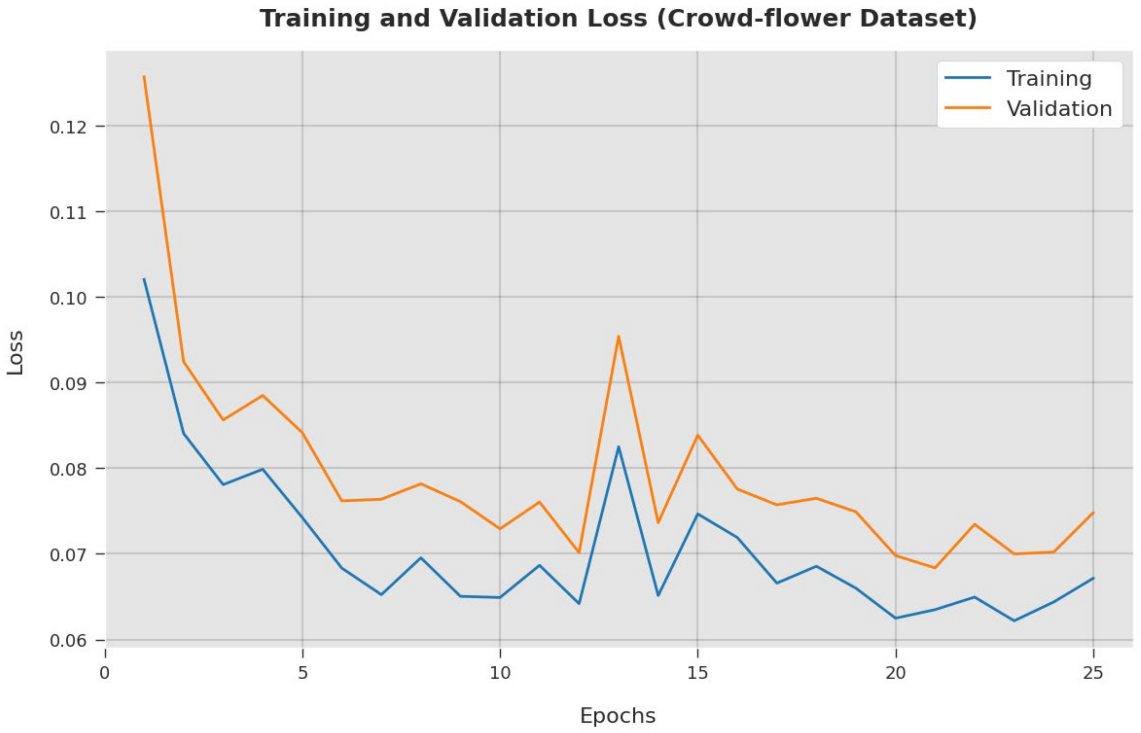

Figure 14. TL and VL analysis of ESGONLP-HSC algorithm under Crowd-flower dataset.

The above-mentioned results and discussion highlighted that the ESGONLP-HSC model has gained an effectual outcome over the other models on hate speech classification.

5. Conclusions

In this study, a new ESGONLP-HSC model has been introduced to identify and classify the occurrence of HS on social media websites. Primarily, the presented ESGONLP-HSC model involves data pre-processing at several stages, such as tokenization, vectorization, etc. Additionally, the Glove technique is applied for the feature extraction process. Next, the ESGO-ABLSTM model is utilized for the classification of social media text into three classes such as neutral, offensive, and hate language. The experimental validation of the ESGONLP-HSC model is carried out, and the results are examined under diverse aspects. The experimentation outcomes reported the promising performance of the ESGONLP-HSC model over recent state-of-the-art approaches.

6. Theoretical and Practical Implications

The proposed model can be tested on large-scale real-time datasets in the future such as Twitter, YouTube, Facebook, news, public meetings, etc. The proposed model can be utilized for hate speech detection in real-time Twitter data and online product reviews. In the future, an ensemble of DL-based fusion models can be integrated to improve the classification performance of the ESGONLP-HSC model.

Author Contributions: Data curation, H.T.H.; Formal analysis, H.T.H.; Investigation, Y.A.; Methodology, Y.A.; Project administration, S.A.-K.; Resources, S.A.-K.; Software, R.F.M.; Supervision, H.M.A.; Validation, H.M.A. and S.H.A.H.; Visualization, S.H.A.H.; Writing—original draft, Y.A.; Writing—review & editing, S.A.-K. and R.F.M. All authors have read and agreed to the published version of the manuscript.

Funding: Deanship of Scientific Research at Umm Al-Qura University for supporting this work by Grant Code: (22UQU4350139DSR03); Deanship of Scientific Research at Najran University for funding this work under the Research Collaboration Funding program grant code (NU/RC/SERC/11/5).

Institutional Review Board Statement: This article does not contain any studies with human participants performed by any of the authors.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data sharing not applicable to this article as no datasets were generated during the current study.

Acknowledgments: The authors would like to thank the Deanship of Scientific Research at Umm Al-Qura University for supporting this work by Grant Code: (22UQU4350139DSR03). Also, the authors are thankful to the Deanship of Scientific Research at Najran University for funding this work under the Research Collaboration Funding program grant code (NU/RC/SERC/11/5).

Conflicts of Interest: The authors declare that they have no conflict of interest.

References

1. García-Díaz, J.A.; Jiménez-Zafra, S.M.; García-Cumbreras, M.A.; Valencia-García, R. Evaluating feature combination strategies for hate-speech detection in spanish using linguistic features and transformers. *Complex Intell. Syst.* **2022**, 1–22. [CrossRef]
2. Kovács, G.; Alonso, P.; Saini, R. Challenges of hate speech detection in social media. *SN Comput. Sci.* **2021**, *2*, 95. [CrossRef]
3. Jahan, M.S.; Oussalah, M. A systematic review of Hate Speech automatic detection using Natural Language Processing. *arXiv* **2021**, arXiv:2106.00742.
4. Alkomah, F.; Ma, X. A Literature Review of Textual Hate Speech Detection Methods and Datasets. *Information* **2022**, *13*, 273. [CrossRef]
5. Al-Makhadmeh, Z.; Tolba, A. Automatic hate speech detection using killer natural language processing optimizing ensemble deep learning approach. *Computing* **2020**, *102*, 501–522. [CrossRef]
6. Pariyani, B.; Shah, K.; Shah, M.; Vyas, T.; Degadwala, S. February. Hate speech detection in twitter using natural language processing. In Proceedings of the 2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV), Tirunelveli, India, 4–6 February 2021; pp. 1146–1152.

7. Perifanos, K.; Goutsos, D. Multimodal Hate Speech Detection in Greek Social Media. *Multimodal Technol. Interact.* **2021**, *5*, 34. [CrossRef]
8. Plaza-del-Arco, F.M.; Molina-González, M.D.; Urena-López, L.A.; Martín-Valdivia, M.T. Comparing pre-trained language models for Spanish hate speech detection. *Expert Syst. Appl.* **2021**, *166*, 114120. [CrossRef]
9. Khan, S.; Kamal, A.; Fazil, M.; Alshara, M.A.; Sejwal, V.K.; Alotaibi, R.M.; Baig, A.R.; Alqahtani, S. HCovBi-caps: Hate speech detection using convolutional and Bi-directional gated recurrent unit with Capsule network. *IEEE Access* **2022**, *10*, 7881–7894. [CrossRef]
10. Khan, S.; Fazil, M.; Sejwal, V.K.; Alshara, M.A.; Alotaibi, R.M.; Kamal, A.; Baig, A.R. BiCHAT: BiLSTM with deep CNN and hierarchical attention for hate speech detection. *J. King Saud Univ.-Comput. Inf. Sci.* **2022**, *34*, 4335–4344. [CrossRef]
11. Husain, F.; Uzuner, O. Investigating the Effect of Preprocessing Arabic Text on Offensive Language and Hate Speech Detection. *Trans. Asian Low-Resour. Lang. Inf. Process.* **2022**, *21*, 73. [CrossRef]
12. Pham, Q.H.; Nguyen, V.A.; Doan, L.B.; Tran, N.N.; Thanh, T.M. November. From universal language model to downstream task: Improving RoBERTa-based Vietnamese hate speech detection. In Proceedings of the 2020 12th International Conference on Knowledge and Systems Engineering (KSE), Can Tho, Vietnam, 12–14 November 2020; pp. 37–42.
13. Robinson, D.; Zhang, Z.; Tepper, J. Hate speech detection on twitter: Feature engineering vs feature selection. In Proceedings of the European Semantic Web Conference, Crete, Greece, 3–7 June 2018; Springer: Cham, Switzerland, 2018; pp. 46–49.
14. Awal, M.R.; Cao, R.; Lee, R.K.W.; Mitrović, S. Angrybert: Joint learning target and emotion for hate speech detection. In Proceedings of the Pacific-Asia Conference on Knowledge Discovery and Data Mining, Virtual Event, 11–14 May 2021; Springer: Cham, Switzerland, 2021; pp. 701–713.
15. Qureshi, K.A.; Sabih, M. Un-compromised credibility: Social media based multi-class hate speech classification for text. *IEEE Access* **2021**, *9*, 109465–109477. [CrossRef]
16. Mohtaj, S.; Schmitt, V.; Möller, S. A Feature Extraction based Model for Hate Speech Identification. *arXiv* **2022**, arXiv:2201.04227.
17. Kumar, D.; Kumar, N.; Mishra, S. QUARC: Quaternion multi-modal fusion architecture for hate speech classification. In Proceedings of the 2021 IEEE International Conference on Big Data and Smart Computing (BigComp), Jeju Island, Korea, 17–20 January 2021; pp. 346–349.
18. Araque, O.; Iglesias, C.A. An ensemble method for radicalization and hate speech detection online empowered by sentic computing. *Cogn. Comput.* **2022**, *14*, 48–61. [CrossRef]
19. Miok, K.; Škrlj, B.; Zaharie, D.; Robnik-Šikonja, M. To BAN or not to BAN: Bayesian attention networks for reliable hate speech detection. *Cogn. Comput.* **2022**, *14*, 353–371. [CrossRef]
20. Nascimento, F.R.; Cavalcanti, G.D.; Da Costa-Abreu, M. Unintended bias evaluation: An analysis of hate speech detection and gender bias mitigation on social media using ensemble learning. *Expert Syst. Appl.* **2022**, *201*, 117032. [CrossRef]
21. Cruz, R.M.; de Sousa, W.V.; Cavalcanti, G.D. Selecting and combining complementary feature representations and classifiers for hate speech detection. *arXiv* **2022**, arXiv:2201.06721. [CrossRef]
22. Yao, Z.; Sun, Y.; Ding, W.; Rao, N.; Xiong, H. Dynamic word embeddings for evolving semantic discovery. In Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining, Marina Del Rey, CA, USA; 2018; pp. 673–681.
23. Hong, G. Relation extraction using support vector machine. In Proceedings of the International Conference on Nat-Ural Language Processing, Jeju Island, Korea, 11-13 October 2005; Springer: Berlin, Germany, 2005; pp. 366–377.
24. Zhang, Z.; Robinson, D.; Tepper, J. Detecting hate speech on Twitter using a convolutionGRU based deep neural network. In Proceedings of the European Semantic Web Conference, Crete, Greece, 3–7 June 2018; Springer: Cham, Switzerland, 2018; pp. 745–760.
25. Kim, Y.; Jernite, Y.; Sontag, D.; Rush, A.M. Character-aware neural language models. In Proceedings of the Thirtieth AAAI Conference on Artifcial Intelligence, Phoenix, AZ, USA, 12–17 February 2016; pp. 1–9.
26. Pennington, J.; Socher, R.; Manning, C.D. Glove: Global vectors for word representation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, 25–29 October 2014; pp. 1532–1543.
27. Liu, G.; Guo, J. Bidirectional LSTM with attention mechanism and convolutional layer for text classification. *Neurocomputing* **2019**, *337*, 325–338. [CrossRef]
28. Dhiman, G.; Kumar, V. Seagull optimization algorithm: Theory and its applications for large-scale industrial engineering problems. *Knowl.-Based Syst.* **2019**, *165*, 169–196. [CrossRef]
29. Caren, T.N.; Jowers, K.; Gaby, S. A social movement online community: Stormfront and the white nationalist movement. In *Media, Movements, and Political Change (Research in Social Movements, Conficts and Change, Volume 33)*; Earl, J., Rohlinger, D.A., Eds.; Emerald Group Publishing Limited: Bingley, UK, 2012; pp. 163–193.
30. Davidson, T.; Warmsley, D.; Macy, M.; Weber, I. Automated Hate Speech Detection and the Problem of Offensive Language. In Proceedings of the 11th International Conference on Web and Social Media (ICWSM), Montreal, QC, Canada, 15–18 May 2017.