Intelligent Automation & Soft Computing
DOI: 10.32604/iasc.2022.027500
Article

Tech Science Press

Glowworm Optimization with Deep Learning Enabled Cybersecurity in Social Networks

Ashit Kumar Dutta[1,*], Basit Qureshi[2], Yasser Albagory[3], Majed Alsanea[4], Anas Waleed AbulFaraj[5] and Abdul Rahaman Wahab Sait[6]

[1]Department of Computer Science and Information Systems, College of Applied Sciences, AlMaarefa University, Ad Diriyah, Riyadh, 13713, Kingdom of Saudi Arabia
[2]Department of Computer Science, Prince Sultan University, Riyadh, 11586, Kingdom of Saudi Arabia
[3]Department of Computer Engineering, College of Computers and Information Technology, Taif University, Taif, 21944, Kingdom of Saudi Arabia
[4]Department of Computing, Arabeast Colleges, Riyadh, 11583, Kingdom of Saudi Arabia
[5]Department of Information Systems, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, 23613, Kingdom of Saudi Arabia
[6]Department of Archives and Communication, King Faisal University, Al Ahsa, Hofuf, 31982, Kingdom of Saudi Arabia
*Corresponding Author: Ashit Kumar Dutta. Email: drashitkumar@yahoo.com
Received: 19 January 2022; Accepted: 23 February 2022

Abstract: Recently, the exponential utilization of Internet has posed several cybersecurity issues in social networks. Particularly, cyberbulling becomes a common threat to users in real time environment. Automated detection and classification of cyberbullying in social networks become an essential task, which can be derived by the use of machine learning (ML) and deep learning (DL) approaches. Since the hyperparameters of the DL model are important for optimal outcomes, appropriate tuning strategy becomes important by the use of metaheuristic optimization algorithms. In this study, an effective glowworm swarm optimization (GSO) with deep neural network (DNN) model named EGSO-DNN is derived for cybersecurity in social networks. The proposed EGSO-DNN technique is mainly intended to identify the presence of cyberbullying on social networking sites. Besides, the EGSO-DNN technique involves different levels of pre-processing to transform the raw data into useful format. In addition, word2vec based feature extraction technique is applied to generate a set of feature vectors. Finally, the DNN model is used for the detection and classification of the DNN model where the hyperparameters of the DNN model are adjusted proficiently by the use of GSO algorithm. In order to ensure the supremacy of the EGSO-DNN technique, a series of simulations were carried out and the results are tested using benchmark datasets. The comparative analysis reported the improvements of the EGSO-DNN technique over the recent approaches maximum prec_n, reca_l, and F1_score of 0.9974, 0.9959, and 0.9966.

Keywords: Social networks; cyberbullying; cybersecurity; deep learning; glowworm swarm optimization

1 Introduction

With the propagation of the Internet, security has become a significant concern [1]. Whereas Web 2.0 offers interactive, easy, access to the online community, also provides an environment for cybercrimes such as cyberbullying. Life annoying cyberbullying experiences amongst younger people were internationally reported, therefore drawing attention to its negative impacts [2]. In the United States, suggestions of cyberbullying are extremely growing and it has formally been recognized as a social threat. The domain of psychology and social sciences were examining the problems of conventional cyberbullying and bullying for a longer time [3]. Several researches have been dedicated to understanding the problems completely and classified the common term "cyberbullying" into certain kinds [4]. The requirement for automatic recognition method for cyberbullying is obvious. A textual instance of offensive language, cyberbullying, and hate speech are mainly recognized by using the conventional machine learning (ML) classifications. In a supervised learning method, pre-processed text input is utilized to train algorithms on a set of datasets. The training set involves data labels and items where the approach learns in a supervised manner [5]. The efficiency of the method is established as testing and validation of the approach on the residual information. The aim is to propose a classification which equally implements well on testing and training information without over-fitting. Classification results estimated by various metrics like recall, accuracy, F1-score, and precision illustrate the efficiency of the classification. Various ML methods need feature extraction from input data. Natural Language Processing (NLP) contain wider application in the field, as researchers are applied various feature extraction method for textual content [6].

The main endeavor includes supervised classification through bag-of-words at character-level depiction [7] using different ML methods like Support Vector Machine (SVM), Random Forest (RF), Linear Regression (LR), extreme gradient boosting (XGBoost), and Naïve Bayes (NB) for cyberbullying identification [8]. The bag-of-words methods extract features of the data set by measuring the existence of words inside a document, disregarding order of the words. Several researchers have presented a new direction of extracting features from the corresponding social networks pertaining to the information. Advancements in method have carried a user-level recognition method in which researcher reporting is utilized for checking a client was involved previously in the act of cyberbullying [9]. Also, current methods are aimed at the user characteristics and background data, like demographic data to characterize malicious users [10]. Deep learning (DL) methods have been utilized for overcoming the limitation of conventional ML, removing the automatic feature extraction phase, and obtaining good outcomes on largescale datasets.

Hani et al. [11] presented the supervised ML technique to detect and prevent cyberbullying. In many classifying were utilized for training and recognizing bullying action. The estimation of the presented method on cyberbullying dataset illustrates that neural network (NN) carries out optimum and attains accuracy. The authors in [12] aimed for exploring this problem with compile a global data set of 37,373 unique tweets on Twitter. Furthermore, 7 ML techniques are utilized such as AdaBoost (ADB), LR, Light Gradient Boosting Machine (LGBM), NB, Stochastic Gradient Descent (SGD), RF, and SVM. All these techniques are estimated utilizing F1-score, precision, accuracy, and recall as efficacy metrics for determining the classifier's detection rate implemented to global data set. In [13], an ML technique was presented for detecting and preventing bullying on Twitter. These are 2 techniques as SVM and NB are utilized to train and test social media bullying content. Combined NB and SVM are capable of detecting the true positive. However, the SVM demonstrates the NB of same work on a similar data set.

In [14], a holistic multi-dimension feature set was established that gets as to social network-based, account individual-based, episode-based, and linguistic content-based cyberbullying features. For testing efficiency of the presented multi-dimension feature set, it can be planned and create cyberbullying recognition methods is an ML platform. 6 distinct ML techniques namely NB, DT, RF, Tree Ensemble,

LR, and SVM are utilized from cyberbullying detection techniques. Elasha et al. [15] appeal the problem of cyberbullying on different online discussion forums under the procedure of social commentary. At this point, supervised ML approaches were utilized for detecting if specific comments are insult, threat, or hate messages. Initially, an ML technique was established with LR, RF, and NB techniques to classifier followed by combine of Voting and AdaBoost classifiers were executed on the established technique for observing that work optimum during this case.

This paper presents a novel EGSO-DNN technique that has been derived for cybersecurity in social networks. Besides, the EGSO-DNN technique involves different levels of pre-processing to transform the raw data into useful format. In addition, word2vec based feature extraction technique is applied to generate a set of feature vectors. Finally, the DNN model is used for the detection and classification of the DNN model where the hyperparameters of the DNN model are adjusted proficiently by the use of GSO algorithm. In order to ensure the supremacy of the EGSO-DNN technique, a series of simulations were carried out and the results are tested using benchmark datasets.

2 The Proposed Model

In this study, an effective EGSO-DNN technique has been derived for the identification of cyberbullying in social networking sites. The EGSO-DNN technique primarily undergoes different levels of pre-processing to transform the raw data into useful format. Also, word2vec based feature extraction technique is applied to generate a set of feature vectors. Furthermore, the DNN model is used for the detection and classification of the DNN model where the hyperparameters of the DNN model are adjusted proficiently by the use of GSO algorithm. Fig. 1 showcases the overall block diagram of proposed EGSO-DNN technique.

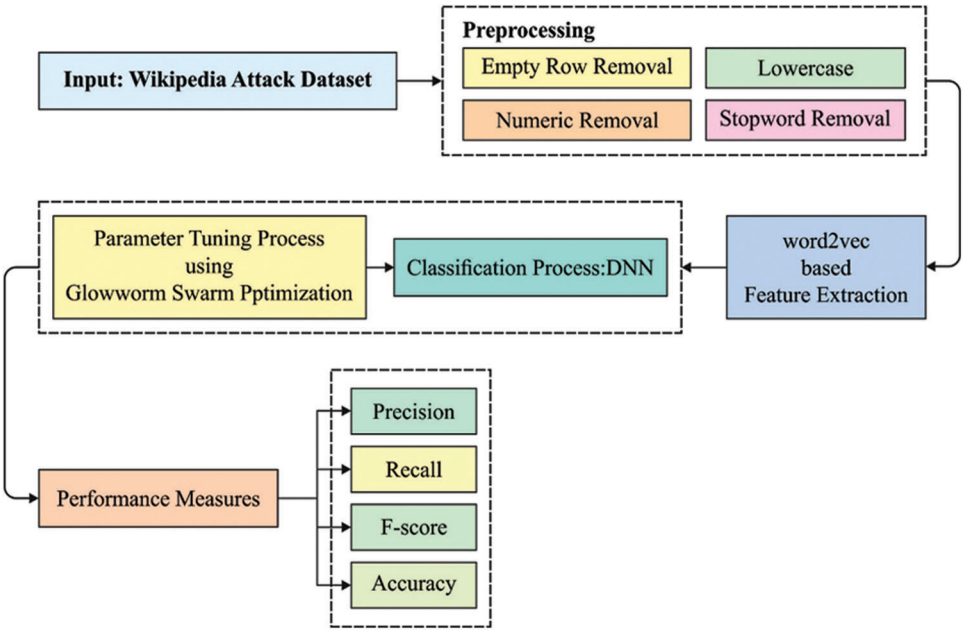

Figure 1: Overall block diagram of EGSO-DNN technique

2.1 Pre-Processing

Since the classifier could not analyze the raw data owing to the inadequacy of avoiding high level human language in a direct way [16]. Therefore, the transformation of text becomes essential for processing the data using classifiers. In this study, data preprocessing take place in different ways such as

- Discarding empty rows,
- Conversion of characters into lowercase,
- Removing punctuation marks,
- Removing special characters,
- Removing numerals,
- Removing stopwords,
- Tokenizing words, and
- Stemmization.

2.2 Word2Vec Based Feature Extraction

The similarity measure (SM) is a function in which the data of 2 ontology entities are utilized as input and real number amongst zero and one has outcome to signify its similarities [17]. In detail, the closer the resultant is to 1, most similar they are; the nearer the resultant is to 0, the lesser similar they are. SM is a vital part of ontology equivalent procedure. Employing distinct SM affects the outcomes of ontology alignment. During this case, it can be utilized 2 types of SMs for calculating the similarity values of 2 entities, for instance, linguistic-based measure and cosine SM utilizing the Word2Vec technique. With concern to the ontology illustration from vector spaces, it represents that a class or property of ontologies is demonstrated from dimensional of vector space. During this case, the dimensional of vector spaces were defined as every class and property from 2 ontologies. The Word2Vec technique was trained to utilize the Wikipedia English corpus. All the entities are signified as a vector from vector spaces, afterward, the similarity of 2 entities are computed utilizing the cosine similarity equation. This equation has been determined as:

$$Cosine\ Similarity(V_{w_1},\ \ V_{w_2}) = \frac{V_{w_1} \cdot V_{w_2}}{V_{w_1} \cdot V_{w_2}}, \tag{1}$$

where V_{w_1} and V_{w_2} are correspondingly the vectors of 2 words w_1 and w_2 and V_{w_1} and V_{w_2} correspondingly, represent its norms. The linguistic similarity amongst 2 words is computed as semantic relation (synonymy and antonymy) that is usually complete utilizing dictionary and list of synonyms. The WordNet, a vocabulary data base which creates semantic network dependent upon the semantic data of words are utilized for calculating similarity. The linguistic similarity of 2 words w_1 and w_2 is 1 if w_1 and w_2 are synonyms from WordNet; the similarity is 0.5 if w_1 and w_2 are hypernym from WordNet; at other times, the similarity has 0. The 2 SMs create 2 similarity matrices, and it can be essential for utilizing an aggregation approach for setting various matrices as to one matrix. During this case, it is experimentally utilized the maximal approach for integrating the SMs, for instance, the superior one of 2 similarity values are chosen as last similarity value that is utilized for ensuring the completeness of alignments.

2.3 DNN Based Cyberbullying Detection Model

During cyberbullying detection and classification process, the features are fed into the DNN model to allot proper class labels. DNN is a kind of NN [18], which is initially partitioned into several two layer models prior to learning the entire approach. Next, the training of the 2-layer NN takes place in a layer-wise manner. At last, the initial weights of the multi-layer NN are comprised of trained 2-layer NN and the entire procedure is known as layer-wise pre-training. The hidden layer of the NN helps in the extraction of features from the input layer because of the abstraction. Therefore, the NN with many hidden layers is effective in network processing as well as fast convergence performance. It is a type of FFNN with several hidden layers and every node kept in the identical hidden layer could utilize the

equivalent non-linear activation function for mapping the feature input from the layer beneath the present node. Fig. 2 depicts the framework of DNN.

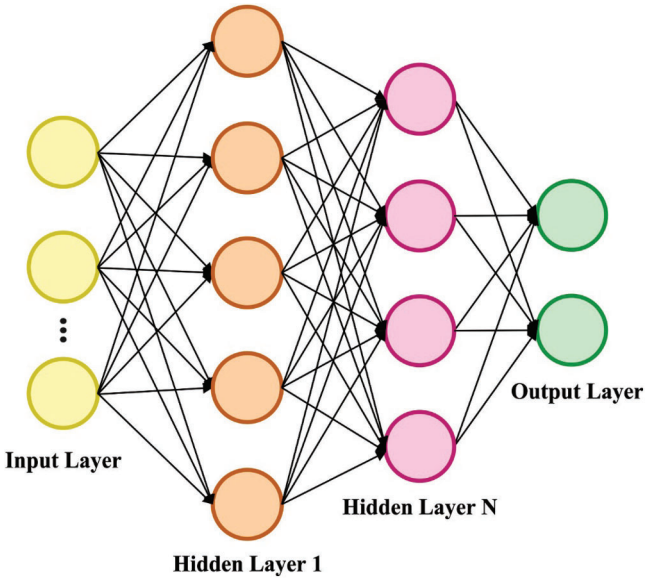

Figure 2: DNN structure

The DNN model exhibits high flexibility owing to the existence of many hidden layers and nodes. In general, the DNN approach is used to classify data. The relativity among the input and output of the DNN model is represented using Eq. (2):

$$v^0 = \text{input},$$
$$v^{l+1} = \rho(z^l(v^l)) \tag{2}$$
$$z^l(v^l) = w^l(v^l) + b^l, \quad 0 \le l < L,$$
$$\text{output} = v^L$$

Based on these mathematical formulations, the end outcome can be obtained by the transformation of the feature vectors in the initial layer v^0 to a processed feature vector v^l via L layers of non-linear transformation. At the time of DNN training process, it is needed to compute the weight matrix w^l and offset vector b^l of lth layer. Using the variation among the target as well as actual output to design a cost function, the training of DNN takes place via the BP technique. The layers in the DNN model are input, output, and hidden layers.

2.4 GSO Based Hyperparameter Tuning Process

The optimal hyperparameter tuning of the DNN model is carried out using the GSO algorithm. The GSO technique is initially devised by Krishnanand et al. [19]. In the GSO algorithm, it is considered that every individual glowworm (GW) has a luciferin level (LL) and a local visibility range (LVR). The LL of a GW is used to determine the light intensity and LVR is used to identify the adjacent GWs that are perceptible to it. The GW selects a neighboring GW having maximum LL compared to themselves in a probabilistic way and fly in the direction of the neighboring GW owing to the light attractively. The LVR can be changed dynamically in order to maintain an appropriate number of neighboring GWs. The GSO algorithm is inspired by the nature of GW and it comprises three major phases as defined below [20]. The

LL upgrade stage determines the LL of all GWs based on the decay of the luminescence and the importance of new position after the execution of the movement in the evolution iteration t. The LL of GW i is can be upgraded using Eq. (3).

$$l_i^{t+1} \leftarrow (1-\rho)l_i^t + \tau f_i^{t+1} \tag{3}$$

where ρ implies decay rate of GW's luminescence and τ denotes an improvement constant. The initial element implies is the perseverance material of luminescence owing to the decay over time, and the next element indicates additive luminescence as a function of f_i^{t+1} representing the objective value determined in the new location of the GW's. At the motion stage of the evolution iterations, the GWs in the swarm carry out a movement process while flying in the direction of the neighboring GW that has maximum LL compared to the incumbent GW, which can be positioned in the local visibility neighborhood represented via radius r_i^t. The likelihood of the GW i attraction to bright GW j at evolution iteration t can be represented using Eq. (4):

$$p_{ij} = \frac{l_j^t - l_i^t}{\Sigma_{k \in N_i^t} l_k^t - l_i^t} \tag{4}$$

where N_i^t indicates a group of GWs exist in the visibility area of GW i at evolution iteration t. When a neighboring GW is chosen, then the present GW carry out a motion process for updating the location using Eq. (5):

$$x_i^{t+1} \leftarrow x_i^t + s\left(\frac{x_j^t - x_i^t}{||x_j^t - x_i^t||}\right) \tag{5}$$

where s implies the movement distance and $\| \bullet \|$ designates denoted vector length. In addition, the movement of GW takes place in s units of distance in the direction of GW j. The upgrading of visibility range takes place in a dynamic way for tuning the visibility radius of every GW for maintaining optimal neighboring count N^*. Therefore, the present neighboring count $N_i^t|$ undergoes comparison with N^* and visibility radius r_i^t can be adjusted using Eq. (6).

$$r_i^{t+1} \leftarrow \min\ \{r_{\max},\ \max\ \{0,\ r_i^t + \eta(N^* - N_i^t)\}\} \tag{6}$$

where $r_{\max}$ indicates higher visibility range and η denotes scaling variable in order to tune N_i^t. So, the r_i^t gets improved when $N_i^t < N^*$, and it gets decreased $N_i^t > N^*$. The possible collection of r_i^t bounds in the interval of $[0,\ r_{\max}]$. The occurrence of $r_i^t = 0$ specifies numerous GWs have resorted to the place of the present GW, whereas $r_i^t = r_{\max}$ disclosed the situation that the present GW is at a large distance closer to neighboring GWs.

Algorithm 1: Pseudocode of GSO Algorithm

Initialization m dimensional

Initialization n glowworm

Assume s as the step size

Consider $x_i(t)$ refers the place of glowworm i at time instant t

Locate agents arbitrarily

$deploy - agents\_randomly$;

for $i = 1$ to n do $\ell_i(0) = \ell_0$

(Continued)

Algorithm 1: (continued)

$r_d^i(0) = r_0$

let highest count of iterations = max_*iter*;

set $t = 1$;

while $(t \leq \max\_iter)$ do:

{

for every glowworm i do:

$\ell_i(t) = (1-\rho)\ l_i(t-1) + \gamma\ Fitness(x_i(t))$;

for all glowworms i do:

{

$N_i(t) = \{j \quad : d_{ij}(t) < r_d^i(t);\ \ell_i(t) < \ell_j(t)\}$;

for all glowworms $j \in N_i(t)$ do:

$p_{ij}(t) = \frac{\ell_j(t) - \ell_i(t)}{\sum p \in N_i(t) \ell_p(t) - \ell_i(t)}$;

$j = choose\_glowworm(\vec{p})$;

$x_i(t+1) = x_i(t) + step\left(\frac{x_j(t) - x_i(t)}{\|x_j(t) - x_i(t)\|}\right)$

$r_d^i(t+1) = \min\ \{r_{sy}, \quad \text{maxm}\ \{0, \quad r_d^i(t) + \beta(n_t - |N_i(t)|)\}\}$;

}

$t \leftarrow t + 1$;

}

3 Experimental Validation

The experimental result analysis of the EGSO-DNN technique take place using the Wikipedia Attack Dataset [21], which includes 115,864 user comments with 13,590 cyberbullying text and 102,274 non-cyber bullying comments.

Fig. 3 demonstrates the set of confusion matrices offered by the EGSO-DNN technique under various BSs. The results show that the EGSO-DNN technique has effectually identified the samples into distinct classes. F0r instance, with BS of 8, the EGSO-DNN technique has recognized 135126 instances into cyberbullying (CB) attack and 101877 instances into Non-CB attack. Likewise, with BS of 16, the EGSO-DNN approach has recognized 135169 instances into CB attack and 101843 instances into Non-CB attack. Meanwhile, with BS of 64, the EGSO-DNN methodology has recognized 135254 instances into CB attack and 101855 instances into Non-CB attack. Eventually, with BS of 128, the EGSO-DNN algorithm has recognized 135155 instances into CB attack and 101890 instances into Non-CB attack. Lastly, with BS of 256, the EGSO-DNN system has recognized 135102 instances into CB attack and 101820 instances into Non-CB attack.

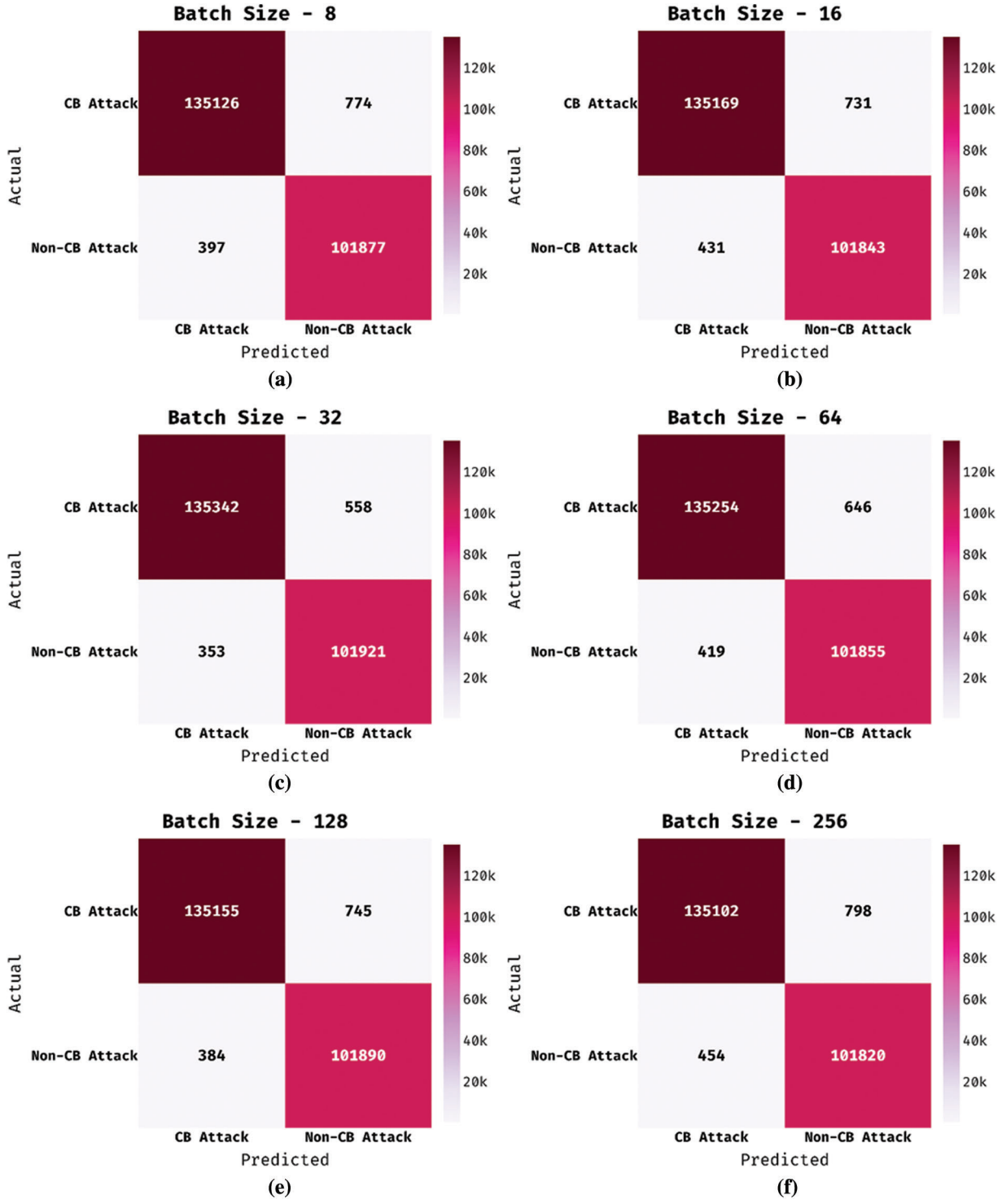

Figure 3: Confusion matrix of EGSO-DNN technique

Tab. 1 and Figs. 4 and 5 offers the overall classification results obtained by the EGSO-DNN technique under distinct batch sizes (BS). The experimental results stated that the EGSO-DNN technique has the

capability of accomplishing enhanced classification performance under every BS. For instance, with BS of 8, the EGSO-DNN technique has resulted to $prec_n$ of 0.9971, $reca_l$ of 0.9943, $accu_y$ of 0.9951, and F_{score} of 0.9957. At the same time, with BS of 16, the EGSO-DNN approach has resulted to $prec_n$ of 0.9968, $reca_l$ of 0.9946, $accu_y$ of 0.9951, and F_{score} of 0.9957. Similarly, with BS of 32, the EGSO-DNN system has resulted to $prec_n$ of 0.9974, $reca_l$ of 0.9959, $accu_y$ of 0.9962, and F_{score} of 0.9966. Along with that, with BS of 64, the EGSO-DNN technique has resulted to $prec_n$ of 0.9969, $reca_l$ of 0.9952, $accu_y$ of 0.9955, and F_{score} of 0.9961. Eventually, with BS of 128, the EGSO-DNN algorithm has resulted to $prec_n$ of 0.9972, $reca_l$ of 0.9945, $accu_y$ of 0.9953, and F_{score} of 0.9958. Lastly, with BS of 256, the EGSO-DNN method has resulted to $prec_n$ of 0.9967, $reca_l$ of 0.9941, $accu_y$ of 0.9947, and F_{score} of 0.9954.

Table 1: Result analysis of EGSO-DNN technique with distinct measures

Batch size	Precision	Recall	Accuracy	F-score
BS-8	0.9971	0.9943	0.9951	0.9957
BS-16	0.9968	0.9946	0.9951	0.9957
BS-32	0.9974	0.9959	0.9962	0.9966
BS-64	0.9969	0.9952	0.9955	0.9961
BS-128	0.9972	0.9945	0.9953	0.9958
BS-256	0.9967	0.9941	0.9947	0.9954
Average	0.9970	0.9948	0.9953	0.9959

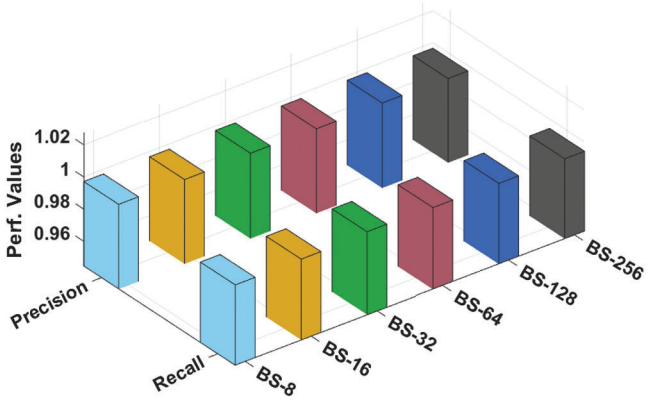

Figure 4: $Prec_n$ and $Reca_l$ analysis of EGSO-DNN technique

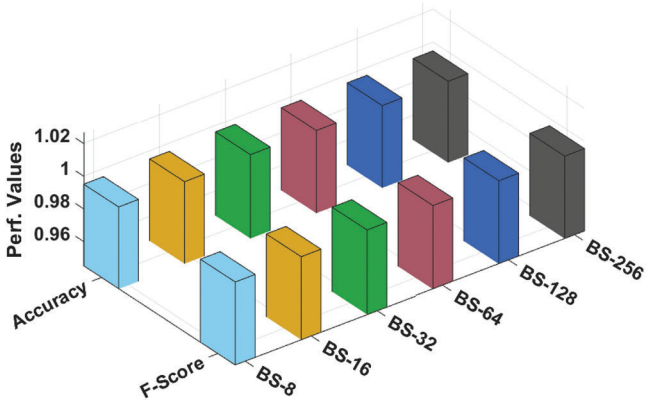

Figure 5: Acc_y and F_{score} analysis of EGSO-DNN technique

Fig. 6 demonstrates the receiver operating characteristic curve (ROC) analysis of the EGSO-DNN system under distinct batch sizes. The figure outperformed that the EGSO-DNN technique has reached higher outcomes with the minimal ROC of 99.9741 under BS of 8.

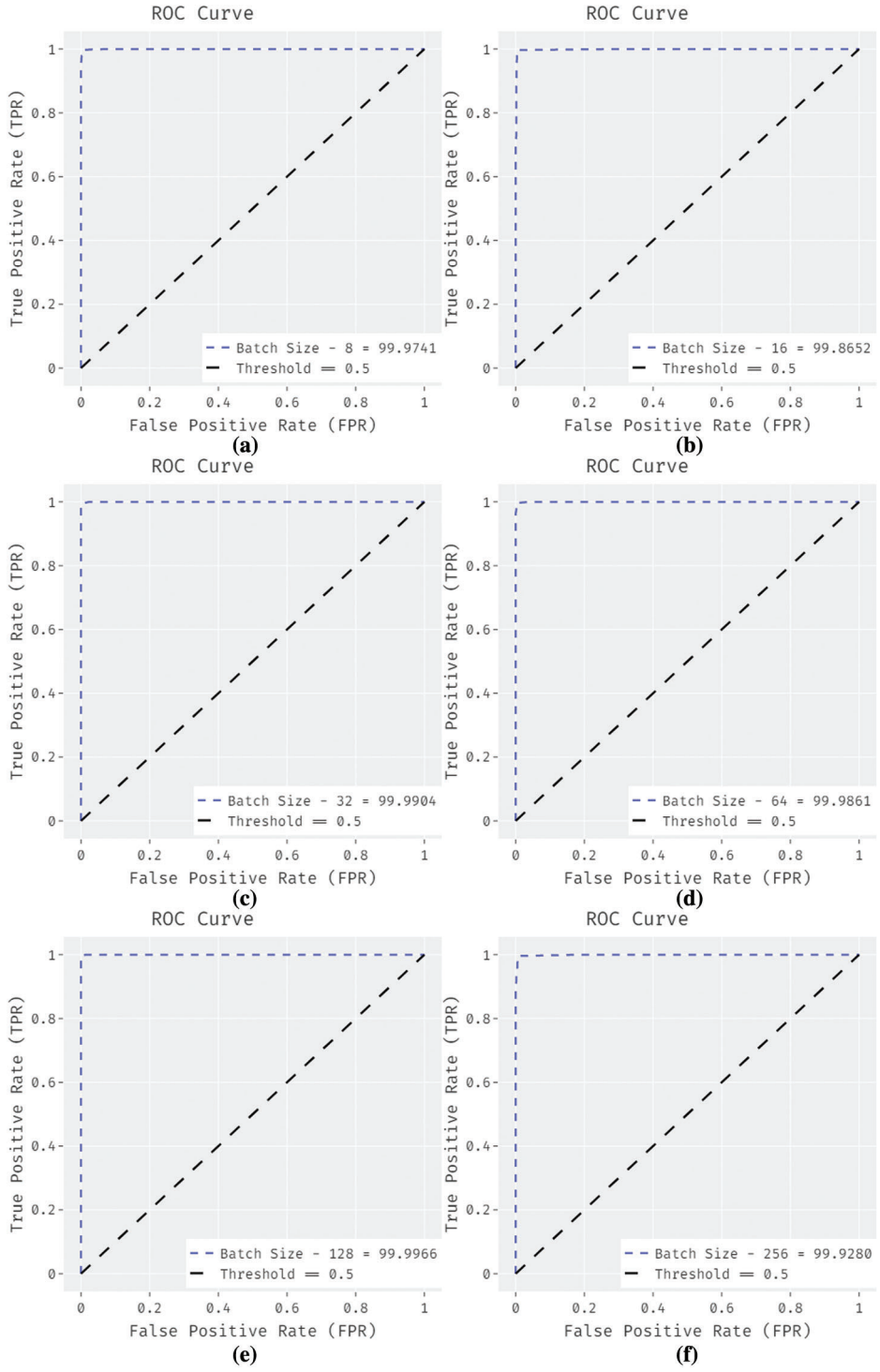

Figure 6: ROC analysis of EGSO-DNN technique with varying batch size

Tab. 2 offers the comparison study of the EGSO-DNN technique with recent methods.

Table 2: Comparative analysis of EGSO-DNN technique with existing methods

Methods	Precision	Recall	Accuracy	F1-score
XG boost	0.9955	0.9536	0.9443	0.9714
NB model	0.9973	0.9627	0.9544	0.9692
SVM model	0.9678	0.9660	0.9380	0.9624
LOGR model	0.9834	0.9611	0.9455	0.9679
CNN model	0.9843	0.9669	0.9560	0.9716
LSTM model	0.9853	0.9685	0.9508	0.9761
GRU model	0.9919	0.9617	0.9546	0.9716
Bi-LSTM model	0.9829	0.9701	0.9688	0.9810
EGSO-DNN	0.9974	0.9959	0.9962	0.9966

The accuracy outcome analysis of the EGSO-DNN approach under BS of 32 is portrayed in Fig. 7. The results outperformed that the EGSO-DNN system has accomplished improved validation accuracy compared to training accuracy. It is also observable that the accuracy values get saturated with the epoch count of 1000.

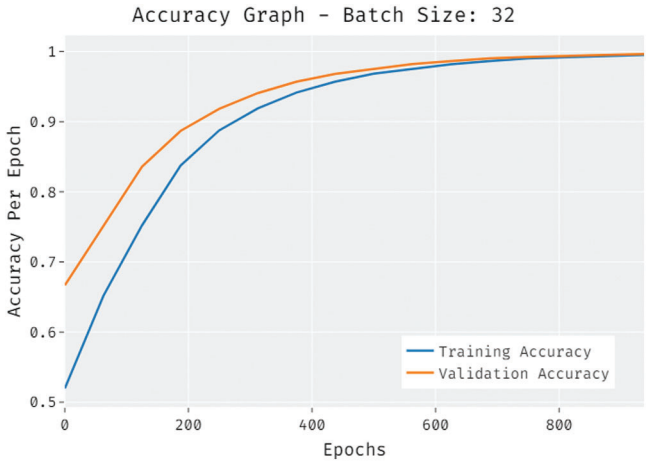

Figure 7: Accuracy graph analysis of EGSO-DNN technique

The loss outcome analysis of the EGSO-DNN technique under BS of 32 is depicted in Fig. 8. The figure revealed that the EGSO-DNN technique has denoted the lower validation loss over the training loss. It is additionally stated that the loss values get saturated with the epoch count of 1000.

Fig. 9 offers the $accu_y$ analysis of the EGSO-DNN technique with existing techniques. The results show that the SVM, XGBoost, and LOGR techniques have shown least outcome with the reduced values of $accu_y$. In line with, the long short term memory (LSTM), NB, gated recurrent unit (GRU), and convolutional neural network (CNN) models have resulted in slightly enhanced values of $accu_y$. Along with that, the bidirectional LSTM (BiLSTM) model has accomplished reasonable performance with the considerably increased values of $accu_y$. However, the EGSO-DNN technique has outperformed the other techniques with the maximum $accu_y$ of 0.9962.

Figure 8: Loss graph analysis of EGSO-DNN technique

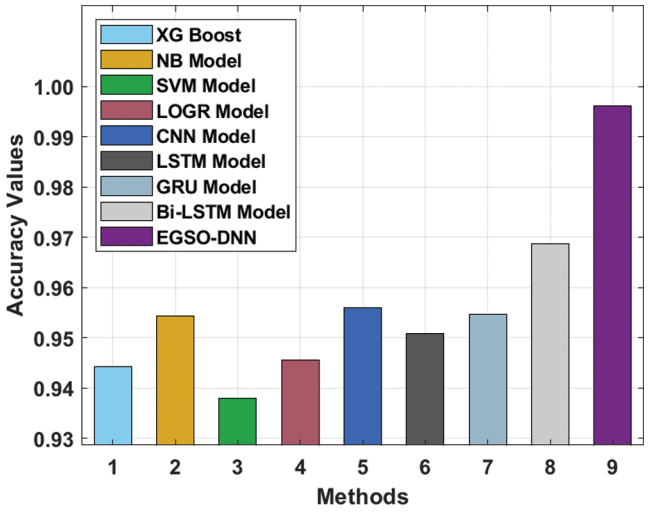

Figure 9: Accuracy analysis of EGSO-DNN technique with existing methods

Fig. 10 provides the $prec_n$, $reca_l$, and $F1_{score}$ analysis of the EGSO-DNN system with existing approaches. The outcomes demonstrated that the SVM, XGBoost, and LOGR methods have illustrated minimal outcomes with the lower values of $prec_n$, $reca_l$, and $F1_{score}$. Also, the LSTM, NB, GRU, and CNN techniques have resulted in slightly higher values of $prec_n$, $reca_l$, and $F1_{score}$. Similarly, the BiLSTM method has accomplished reasonable performance with the considerably increased values of $prec_n$, $reca_l$, and $F1_{score}$. At last, the EGSO-DNN algorithm has exhibited the other techniques with the maximum $prec_n$, $reca_l$, and $F1_{score}$ of 0.9974, 0.9959, and 0.9966.

From the above results and discussion, it is confirmed that the EGSO-DNN technique has accomplished enhanced performance over the other techniques.

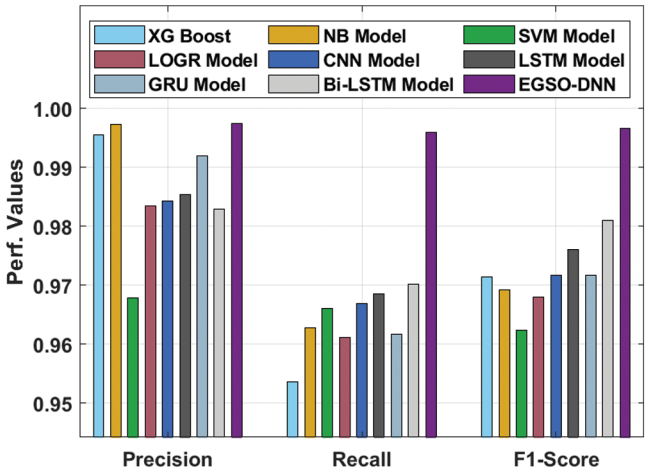

Figure 10: Comparative analysis of EGSO-DNN technique with existing approaches

4 Conclusion

In this study, a novel EGSO-DNN technique has been derived for cybersecurity in social networks. The proposed EGSO-DNN technique is mainly intended to identify the presence of cyberbullying on social networking sites. Besides, the EGSO-DNN technique involves different levels of pre-processing to transform the raw data into useful format. In addition, word2vec based feature extraction technique is applied to generate a set of feature vectors. Finally, the DNN model is used for the detection and classification of the DNN model where the hyperparameters of the DNN model are adjusted proficiently by the use of GSO algorithm. In order to ensure the supremacy of the EGSO-DNN technique, a series of simulations were carried out and the results are tested using benchmark datasets. The comparative analysis reported the improvements of the EGSO-DNN technique over the recent approaches in several aspects. In future, hybrid DL models can be used to enhance the overall performance.

Acknowledgement: The authors deeply acknowledge the Researchers supporting program (TUMA-Project-2021-27) Almaarefa University, Riyadh, Saudi Arabia for supporting steps of this work. The authors would like to acknowledge the support of Prince Sultan University for paying the Article Processing Charges (APC) of this publication.

Funding Statement: This research was supported by the Researchers Supporting Program (TUMA-Project-2021-27) Almaarefa University, Riyadh, Saudi Arabia. Taif University Researchers Supporting Project Number (TURSP-2020/161), Taif University, Taif, Saudi Arabia.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] M. F. L. Vizcaíno, F. J. Nóvoa, V. Carneiro and F. Cacheda, "Early detection of cyberbullying on social media networks," *Future Generation Computer Systems*, vol. 118, pp. 219–229, 2021.

[2] C. Cai, L. Li and D. Zeng, "Detecting social bots by jointly modeling deep behavior and content information," in *Proc. of the 2017 ACM on Conf. on Information and Knowledge Management*, Singapore, pp. 1995–1998, 2017.

[3] A. Aggarwal, A. Rani, P. Sharma, M. Kumar, A. Shankar *et al.,* "Prediction of landsliding using univariate forecasting models," *Internet Technology Letters*, vol. 5, no. 1, pp. e209, 2022.

[4] S. Srivastava, S. Saxena, R. Buyya, M. Kumar, A. Shankar *et al.,* "CGP: Cluster-based gossip protocol for dynamic resource environment in cloud," *Simulation Modelling Practice and Theory*, vol. 108, pp. 102275, 2021.

[5] Z. Zhao, P. Resnick and Q. Mei, "Enquiring minds: Early detection of rumors in social media from enquiry posts," in *Proc. of the 24th Int. Conf. on World Wide Web*, Florence Italy, pp. 1395–1405, 2015.

[6] D. E. Losada, F. Crestani and J. Parapar, "ERISK 2020: Self-harm and depression challenges," in *European Conf. on Inf. Retrieval*, Bethesda MD, USA: Springer, pp. 557–563, 2020. https://doi.org/10.1007/978-3-030-45442-5_72.

[7] N. Lu, G. Wu, Z. Zhang, Y. Zheng, Y. Ren *et al.,* "Cyberbullying detection in social media text based on character-level convolutional neural network with shortcuts," *Concurrency and Computation: Practice and Experience*, vol. 32, no. 23, pp. 1–11, 2020.

[8] C. Iwendi, G. Srivastava, S. Khan and P. K. R. Maddikunta, "Cyberbullying detection solutions based on deep learning architectures," *Multimedia Systems*, pp. 1–14, 2020. https://doi.org/10.1007/s00530-020-00701-5.

[9] V. Balakrishnan, S. Khan and H. R. Arabnia, "Improving cyberbullying detection using twitter users' psychological features and machine learning," *Computers & Security*, vol. 90, pp. 101710, 2020.

[10] S. Mahbub, E. Pardede and A. S. M. Kayes, "Detection of harassment type of cyberbullying: A dictionary of approach words and its impact," *Security and Communication Networks*, vol. 2021, pp. 1–12, 2021.

[11] J. Hani, M. Nashaat, M. Ahmed, Z. Emad, E. Amer *et al.,* "Social media cyberbullying detection using machine learning," *International Journal of Advanced Computer Science and Applications*, vol. 10, no. 5, pp. 703–707, 2019.

[12] A. Muneer and S. M. Fati, "A comparative analysis of machine learning techniques for cyberbullying detection on twitter," *Future Internet*, vol. 12, no. 11, pp. 187, 2020.

[13] R. R. Dalvi, S. B. Chavan and A. Halbe, "Detecting a twitter cyberbullying using machine learning," in *2020 4th Int. Conf. on Intelligent Computing and Control Systems (ICICCS)*, Madurai, India, pp. 297–301, 2020.

[14] Y. Liu, P. Zavarsky and Y. Malik, "Non-linguistic features for cyberbullying detection on a social media platform using machine learning," in *Int. Symp. on Cyberspace Safety and Security*, Guangzhou, China, pp. 391–406, 2019.

[15] F. Elasha, S. Shanbr, X. Li and D. Mba, "Prognosis of a wind turbine gearbox bearing using supervised machine learning," *Sensors*, vol. 19, no. 14, pp. 3092, 2019.

[16] C. Raj, A. Agarwal, G. Bharathy, B. Narayan and M. Prasad, "Cyberbullying detection: Hybrid models based on machine learning and natural language processing techniques," *Electronics*, vol. 10, no. 22, pp. 2810, 2021.

[17] X. Xue, H. Wang, J. Zhang, Y. Huang, M. Li *et al.,* "Matching transportation ontologies with word2vec and alignment extraction algorithm," *Journal of Advanced Transportation*, vol. 2021, pp. 1–9, 2021.

[18] W. Samek, A. Binder, G. Montavon, S. Lapuschkin and K. R. Muller, "Evaluating the visualization of what a deep neural network has learned," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 11, pp. 2660–2673, 2017.

[19] K. N. Krishnanand and D. Ghose, "Glowworm swarm optimization for simultaneous capture of multiple local optima of multimodal functions," *Swarm Intelligence*, vol. 3, no. 2, pp. 87–124, 2009.

[20] P. Y. Yin, P. Y. Chen, Y. C. Wei and R. F. Day, "Cyber firefly algorithm based on adaptive memory programming for global optimization," *Applied Sciences*, vol. 10, no. 24, pp. 8961, 2020.

[21] E. Wulczyn, N. Thain and L. Dixon, "Ex machina: Personal attacks seen at scale," in *Proc. of the 26th Int. Conf. on World Wide Web*, Perth, Australia, pp. 1391–1399, 2017.