Hyper-tuned Swarm Intelligence Machine Learning-based Sentiment Analysis of Social Media

Nitesh Sureja
Department of CSE, KSET, Drs. Kiran & Pallavi Patel Global University (KPGU), Vadodara, India
dir.kset@kpgu.ac.in

Nandini Chaudhari
Department of IT, KSET, Drs. Kiran & Pallavi Patel Global University (KPGU), Vadodara, India
hod.it.kset@kpgu.ac.in

Priyanka Patel
Department of CSE, KSET, Drs. Kiran & Pallavi Patel Global University (KPGU), Vadodara, India
priyankapatel.cse.kset@kpgu.ac.in

Jalpa Bhatt
Department of CSE, KSET, Drs. Kiran & Pallavi Patel Global University (KPGU), Vadodara, India
jalpabhatt.cse.kset@kpgu.ac.in

Tushar Desai
Department of CSE, KSET, Drs. Kiran & Pallavi Patel Global University (KPGU), Vadodara, India
tushardesai.cse.kset@kpgu.ac.in

Vruti Parikh
Department of CSE, KSET, Drs. Kiran & Pallavi Patel Global University (KPGU), Vadodara, India
vrutiparikh.cse.kset@kpgu.ac.in

Received: 13 May 2024 | Revised: 25 May 2024 | Accepted: 29 May 2024

Licensed under a CC-BY 4.0 license | Copyright (c) by the authors | DOI: https://doi.org/10.48084/etasr.7818

ABSTRACT

Natural Language Processing (NLP) uses Sentiment Analysis (SA) to determine text sentiment. SA is often used on text datasets to assess consumer demands, the sentiment of the customer for a product, and brand monitoring. Deep Learning (DL) is a subset of Machine Learning (ML) that mimics how humans learn. In this work, the Deep Learning Reptile Search Algorithm (SA-DLRSA) model is introduced for accurate automatic SA. The SA-DLRSA model utilizes Word2Vec word embedding to reduce language processing that is dependent on data pre-processing. The SA-DLRSA model utilizes SVM, CNN, RNN, BiLSTM, and BERT models for sentiment classification. Choosing the optimal hyperparameters is crucial for determining the model's architecture, functionality, performance, and accuracy. The Reptile Search Algorithm (RSA) is employed to find the best optimal hyperparameters to improve classification. A derived balanced dataset based on the tweets related to bitcoins was employed as a training dataset, which contains three sentiments, namely "neutral", "positive", and negative". The collection has 7 columns and 50058 rows, consisting of 21938 neutral, 22937 positive, and 5183 negative tweets. Precision, accuracy, recall, and F1 Score metrics were used to evaluate the effectiveness of the proposed approach. The results showed that the BERT and BiLSTM classifiers achieved superior performance in classifying sentiments in the tweets achieving accuracies of 99% and 98%, respectively. Due to the promising results of the proposed approach, it is anticipated to be used in solutions to social media problems, such as hate speech detection and emotion detection.

Keywords-machine learning; natural language processing; sentiment analysis; reptile search algorithm

I. INTRODUCTION

As internet and handheld gadget use increased, so does the use of online social platforms like Twitter, Instagram, and Facebook [1]. Nowadays, people want to share their product reviews, feedback, and news through social media [2]. The free features of social media allow real-time detection of new developments or changes in reality, which can be shared or developed and immediately broadcast as content [3]. Studies on social media content analysis are conducted regularly. Thus, smart techniques for analysis of sentiments that convert raw online social platform user facts into meaningful information are needed. SA is AI's most common and hardest problem. It uses methods to recognize phycological information like attitudes, opinions, and sentiments in text and subtext on blogs, news, and social media [4]. SA identifies negative, positive or neutral sentiments in a statement, clause, paragraph, or document. It is used commercially to recognize client satisfaction/dissatisfaction with a product [5]. SA on survey responses and social media discussions may help firms identify issues and adapt their products to client needs [6]. Effective SA techniques are needed as massive amounts of non-uniform data are shared or created in real time [7].

Machine Learning (ML) has been widely used for SA to classify and detect sentiments from social media [9]. ML-related SA uses statistical techniques like word embedding to vectorize a word and construct a digital sentence using ML or DL. Authors in [10] suggest a subject-based SA model based on DL. The suggested method uses topic-modeling techniques known as RLSI to extract subjects at the statement level, and then uses the topic modeling in the Long Short-Term Memory (LSTM) to conduct SA. Authors in [11] predicted sentiments on Chinese social media using a BiLSTM model based on Valence-Arousal information. The experimental findings demonstrate the superiority of the proposed strategy over existing methods for Valence-Arousal or Chinese short text sentiment prediction. Authors in [12] used a customized DL approach with increased word embedding to improve sentiment categorization and construct an LSTM system. They also proposed an ensemble technique that compares the baseline classifier to more advanced SA classifiers. Authors in [13] used a pretrained language algorithm named BERT to create word embeddings. Additionally, three parallel DCNN layers with an enhanced pooling layer helped fine-tune the algorithm. The proposed BERT-DCNN approach minimizes dimensionality and increases associated dimensions to hold data loss. Authors in [14] introduced a method for SA using network embeddings. An encoder was used to study combined representations of social bonds preserving structural and attribute closeness and a VAE based on multi-view association learning was suggested to combine the sentiment polarity network with the combined representation. Authors in [15] presented a novel DL method for the SA in Arabic. They used a single-layer CNN for the extraction of features. LSTM layers were used to handle the dependency over time and an SVM classifier was used for the classification using features derived by the CNN and the LSTM. Authors in [16] studied how contextual embedding models work in BERT and in Arabic language. Authors in [17] investigate the aspect-based SA in English and Arabic. Two DL-based approaches were used to solve these problems, i.e. aspect-based SA and GRU. They predicted the number of software faults using different deep learning techniques.

In this work, ML combined with the Reptile Search Algorithm (SA-DLRSA) is used to perform social media SA. For sentiment classification, methods like SVM, CNN, RNN, BiLSTM, and BERT are used. RSA is used here to optimize the hyperparameters.

A Convolution Neural Network (CNN) is a popular image categorization Deep Neural Network (DNN) approach [19]. A CNN contains a three-dimensional arrangement of neurons instead of the standard two-dimensional array. The first layer is called a convolutional layer. Each neuron in the convolutional layer only processes the information from a small part of the visual field. Input features are taken in batch-wise like a filter. The output of the convolution layer goes to a fully connected neural network for classifying the images. Filters are used to extract certain parts of the image. In MLP, the inputs are multiplied with weights and fed to the activation function. Convolution uses RELU activation function followed by softmax. CNNs show very effective results in image and video recognition, semantic parsing, and paraphrase detection.

In Recurrent Neural Networks (RNNs), each output element is evaluated as a function of previous elements of the output [20]. All the output elements are calculated by applying the same rule of updating the earlier outcomes.

The Bidirectional LSTB (BiLSTM) allows to efficiently use knowledge from both previous and subsequent contexts. It also effectively represents forward and backward word and sentence sequences [21]. In essence, the BiLSTM architecture uses an additional LSTM layer to reverse information flow. So, in the additional LSTM layer input, the sequences are reversed. Both LSTM layer outputs are combined by averaging, summing, multiplying, or concatenating.

Support Vector Machines (SVMs) use supervised learning for regression, outlier detection, and classification. Due to their decision boundary selection method, SVMs differ from other classification techniques. SVMs find the decision boundary that minimizes the distance between the nearest data points from all classes. [22]. A simple SVM classifier creates a linear border between classes. Thus, there may be an infinite number of lines to analyze. The linear SVM algorithm selects the best hyper plane for classifying data points. The algorithm chooses the line that divides the data and optimizes distance from nearby locations.

Bidirectional Encoder Representations from Transformers (BERT) was created for pre-training deep bi-directional representations [23]. A pre-trained BERT model can be fine-tuned for diverse tasks by adding an output layer without changing architecture. BERT effectively captures information from both preceding and following tokens in deep bidirectional models. It differs from prior methods that examined a text sequence from the left, right, or both sides. Language models better understand linguistic context with a bidirectional model. BERT generates language models using the transformer architecture. BERT's bidirectional nature needs the encoder mechanism, making the decoder superfluous. The transformer model is bi-directional or non-directional. This allows the model to learn contextual information about a word from its preceding and succeeding words, regardless of sentence position. BERT uses Masked LM (MLM) and Next Sentence Prediction (NSP) to solve context learning problems. About 15% of each sequence's words are replaced with a token in the MLM. Next, a forecasting session is used to determine the masked word's underlying word using the other unobscured words in sequence. The encoder output is modified by adding a classification layer. Next, the result vectors are converted into vocabulary dimensions. The output vectors are multiplied by the embedding matrix. The final step is calculating the vocabulary word probabilities. This technique only predicts hidden values while not overlooking revealed values. The NSP method forecasts the next sentence using a primary pair of sentences from the original material, unlike the initial approach. About 50% of the primary text's input sentences are sequentially related to the next, while the other 50% are randomly picked and unrelated. The BERT model reduces loss via MLM and NSP.

BERT is a multilayer transformer network pre-trained for a range of self-supervised applications. BERT is only suitable for short text due to its input length limitation. This problem was solved in [24] with their innovative method, which uses BERT to summarize lengthy documents. They employed the technique of breaking up a lengthy document into smaller sections, each of which has a single sentence. Essentially, the plan was to use BERT's sentence embeddings and then build an encoder-decoder model on top of it. Authors in [25] integrated BERT with BiGRU, an RNN that captures text dependencies, to extract salient information. The BiGRU network receives this salient information from BERT which in turn helps grasp the document's context. This architecture helps build ranking image representations and complicated relationships [26, 27].

Effective SA still faces several unresolved challenges, such as the optimal selection of hyperparameter values, low accuracy, computational complexity, time constraints, and a lack of natural datasets. Various studies have proposed diverse methods for analyzing and categorizing sentiments. The objective of this study is to address the aforementioned gap by utilizing a derived bitcoin dataset. This will be accomplished by employing five ML classifiers: SVM, RNN, CNN, BiLSTM, and BERT. This paper introduces RSA, a swarm intelligence algorithm. This algorithm is utilized to optimize the hyperparameters of the ML models in order to accurately identify and categorize views. The objective of this study is to assist different applications and organizations that depend on accurate sentiment classification to make unbiased and optimal decisions. The main contributions of the current study are:

- A novel approach for SA is presented.
- Five different ML models improved with RSA are studied.
- RSA is used to tune the hyperparameters of the models.
- Evaluation measures such as recall, F1 Score, accuracy, and precision are used.
- The outcomes of the ML models are compared.

II. MATERIALS AND METHODS

This work built a novel SA-DLRSA model to identify and classify social media sentiments. The SA-DLRSA approach mostly transforms raw social media text into meaningful information. Word2Vec word embedding scheme is used to reduce language processing dependent on data-pre-processing.

A. Data Preparation

A derived dataset achieved by merging two datasets based on tweets on Bitcoins is utilized. The two datasets used are taken from Kaggle [34, 35], with [34] being the main dataset and the data from [35] were refined to match to those in [34]. We have removed some columns from both datasets to finally get a balanced and meaningful dataset to work with. Using Python's concatenating method, the two datasets were merged. The collection as a whole has 7 columns and 50058 rows. The missing numbers were taken out, and mapping was conducted. Special letters, punctuation, numbers, symbols, and hash tags were also taken out of the model dataset. The sentiment collection has 21938 neutrals, 22937 positives, and 5183 negatives. The goal of preparing the data is to remove undesired and noisy data. Tokenization, case conversion, punctuation removal, and stop word removal were conducted, and SA was simplified by reducing word-to-root or stem formation.

B. Word Embedding

Due to their accuracy, ANN-based NLP models are becoming more popular. But the majority of NLP techniques are slow when evaluating large datasets and require word embedding for text datasets. We used Wod2Vec word embedding scheme to improve the proposed systems' performance and processing speed. Language is a system of unusual occurrences that is so complicated and variable that NLP can't model every possible outcome. So, to analyze non-adjacent words and skip some words, Wod2Vec is used in this work.

C. Hyperparameter Optimization

During the training of the ML models, it is necessary to specify a distinct collection of hyperparameters for each dataset and model. Hyperparameters are external configuration variables employed to govern the training of ML models. Choosing the appropriate hyperparameters is crucial for determining the model's architecture, functionality, performance, and accuracy. Unfortunately, there is a lack of established guidelines for the most effective hyperparameters and their ideal or default values. It is necessary to conduct experiments in order to determine the most effective combination of hyperparameters. This process is referred to as hyperparameter tuning. Hyperparameter tuning can be done either manually or automatically. This research employs the RSA [28] for hyperparameter tuning. Hyperparameter tuning is an iterative process where we experiment with various combinations of parameters and values. To begin, we establish a goal variable, such as accuracy, as the primary metric with the objective of either maximizing or minimizing this variable. In this study, we have emphasized on hyperparameters such as learning rate, optimizer, activation function, batch size and number of epochs. We used the RSA to optimize them in an iterative manner based on a fitness value assigned to each hyperparameter. During each iteration of the RSA the values of the parameters' current best combination are updated. Then these parameter values are used by the all ML models to classify the tweets. The results achieved are compared with the best achieved so far and the best among them are stored as the current best. In this way, the process continues utnil the termination criteria of the RSA and the final best will be stored.

The RSA emulates the predatory behavior of crocodiles in their native environment [28]. The key difference between the RSA algorithm and other optimization algorithms is that RSA has a unique method to update the search-agent locations using four new methods. For instance, the act of surrounding is conducted by high-walking or belly-walking, and the crocodiles communicate or collaborate to perform hunting. RSA has proved itself better than other Swarm Intelligence (SI) algorithms for finding the optimal solutions for different kinds of problems [29-32]. RSA operates through three distinct phases, analyzed below.

1) Initialization Phase

RSA creates the initial solution randomly and uses (1) for the creation of the initial solution defined as: A_i^1 represents the ith initial individual, while LowerBound (LOB) and UpperBound (UPB) represent the lower and upper borders of the search area, respectively.

$$A_i^1 = LOB + ran \times (UPB - LOB) \quad (1)$$

2) Encircling Phase (Exploration)

Crocodiles engage in high and sprawling walks within the global search phase. The search technique in RSA is defined by the current number of cycles. RSA exhibits a high walk when it is less than or equal to 0.25 *IT*. When the input is less than or equal to 0.25 *IT*, and greater than 0.25 *IT*, the RSA executes a sprawl walk. The computational models that describe the mechanism are outlined as follows:

$$A_i^{t+1} = \begin{cases} A_{best}^{it} - \eta_i \times \alpha - R_i^{it} \times ran, \\ \qquad it \le \frac{IT}{4} \\ A_{best}^{it} \times A_{ran}^{it} \times EVS \times ran, \\ \qquad it \le \frac{IT}{4} \text{and it} > \frac{IT}{4} \end{cases} \quad (2)$$

$$\eta_i = A_{best}^{it} \times B_i \quad (3)$$

$$R_i = \frac{A_{best}^{it} - A_i^{it}}{A_{best}^{it} + \varepsilon} \quad (4)$$

$$EVS = 2 \times r_1 \times \left(1 - \frac{1}{IT}\right) \quad (5)$$

$$B_i = \beta + \frac{A_i^{it} - M(A_i^{it})}{A_{best}^{it} \times (UpperBound - LowerBound) + \varepsilon} \quad (6)$$

$$M = \frac{1}{n} \sum_{j=1}^{n} A_i \quad (7)$$

where A_{best}^{it} represents the most current optimal solution, *it* represents the current iteration, *IT* represents the maximum iterations, α is a constant that controls the rate of exploration and has a value of 0.1, A_{ran}^{it} is an randomly chosen individual, ε is a minimal value that prevents the denominator from being equal to 0, r_1 and *ran* are random numbers in the range [-1, 1] and [0, 1], respectively, and β represents a constant and has a value of 0.1. The hunting operator in the ith solution is represented as η_i and is computed using (3). Evolutionary Perception (EVS) is a stochastic variable that ranges from 2 to -2. It quantifies the likelihood of observing declining values during the repetitions, as determined by (5). The term *Bi* represents the discrepancy between the location of the best solution found so far and the location of the current solution. It is derived by (6). *M* represents the average positions of the *i*th solution, determined using by (7).

3) The Exploitation Phase (Hunting)

RSA crocodiles employ two distinct tactics for foraging: hunting cooperation and coordination. While the rate of *it* is less than 0.75 *IT* and greater than or equal to 0.5 *IT*, the RSA engages in hunting coordination. When the rate reaches or exceeds 0.75 *IT*, the RSA employs a hunting cooperation tactic. The updating of the position during the hunting phase is carried out in the following manner:

$$A_i^{it+1} = \begin{cases} A_{best}^{it} - B_i \times ran, it \le \frac{IT}{4} \; and \; it > \frac{IT}{2} \\ A_{best}^{it} \times \eta_i \times \varepsilon - R_i^{it} \times ran, it \le IT \; and \; it > \frac{3IT}{4} \end{cases} \quad (8)$$

The initial population in the search space is generated randomly by RSA, and the number of iterations determines the search strategy. RSA pseudocode can be seen in Figure 1.

4) Fitness Function for RSA

The RSA method defines and uses a fitness function to improve the performance of a classifier. It assigns a positive number to the solutions that perform well. The fitness function in this research was the classification error rate reduction, as indicated in (9).

$$Fitn(x_i) = \frac{\text{Total wrongly classified instances}}{\text{Total instances}} * 100 \quad (9)$$

RSA evaluates the fitness assigned to each hyperparameter in each iteration and updates the best until the termination of the execution.

III. RESULTS AND DISCUSSION

A. Assessment Indicators

The prediction models were assessed using the 10-fold cross-validation technique, with evaluation metrics including F1-score, recall, precision, and accuracy:

$$\text{Precision } (P) = TP/(TP + FP) \quad (10)$$

$$\text{Recall } (R) = TP/(TP + FN) \quad (11)$$

$$\text{F1 Score} = (2 * P * R)/(P + R) \quad (12)$$

$$\text{Accuracy } (A) = (TN + TP)/(TN + FP + TP + FN) \quad (13)$$

TP (true positive) denotes the count of positive cases that are correctly classified as positive. True Negative (*TN*) is the number of negative events that are accurately identified as negative. False Positive (*FP*) refers to the number of negative cases that are mistakenly classified as positive and False Negative (*FN*) represents the count of positive cases that are incorrectly classified as negative.

```
Initialize parameters of RSA. Randomly create starting population
While it < IT
   Compute the Fitness of all individuals (solutions)
  Search Best individual so far
   Do update the EVS using Equation 2
    For every crocodile A_i do
      Do Update R, η, and other values using Equations 3, 4 and 6
        If it < 0.25IT
            Compute new location A_i using Equation 2
        Else if it ≤ 0.5 IT and it > 0.25IT
            Compute new location A_i using Equation 2
        Else if it ≤ 0.75T and t > 0.5IT
            Compute new location A_i using Eq. (8)
         Else
            Compute new location A_i using Eq. (8)
        End if
   Compute fitness and choose best one
   End for
it = it + 1
End while
Return output (best fitness and location)
```

Fig. 1. Reptile search algorithm.

B. Experimental Results

Tables I shows the optimal parameter settings achieved after hyper parameter tuning for the all models used in this study. Table II shows the summary of the F1-score, accuracy, precision, and recall of the studied models. The actual total positive, neutral, and negative tweets are 22937, 21938, and 5183 respectively (Figure 2). Figures 4-8 show the confusion matrices of the considered algorithms.

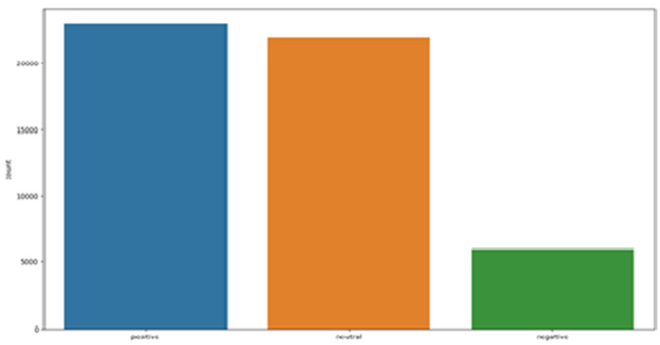

Fig. 2. Tweet count.

TABLE I. HYPER TUNED PARAMETER SETTING OF THE MODELS

Parameter	SVM	CNN	RNN	BiLSTM	BERT
Activation function	Softmax				
Batch size	128				
Optimizer	Adam				
Epoch number	10				
Learning rate	----	----	0.001	-------	1×10^{-5}

TABLE II. RESULT SUMMARY

Model	Accuracy	Precision	Recall	F1 Score
SVM	96%	95%	93%	94%
RNN	97%	96%	94%	97%
CNN	97%	96%	93%	96%
BiLSTM	98%	98%	92%	97%
BERT	99%	99%	95%	98%

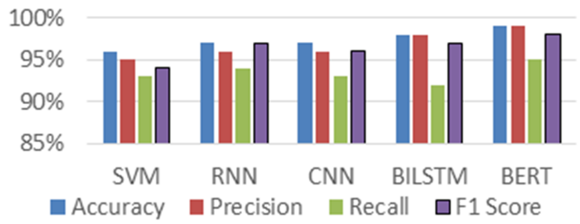

Fig. 3. Result summary.

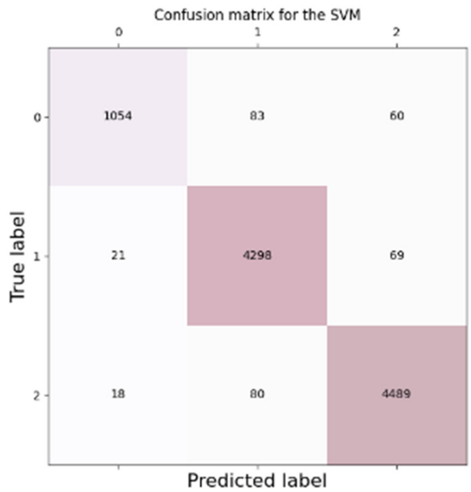

Fig. 4. Confusion Matrix (SVM).

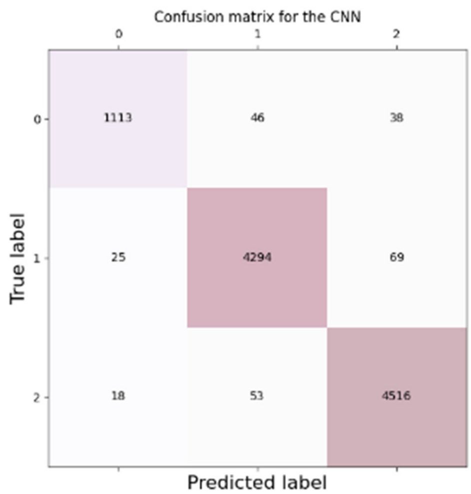

Fig. 5. Confusion Matrix (CNN).

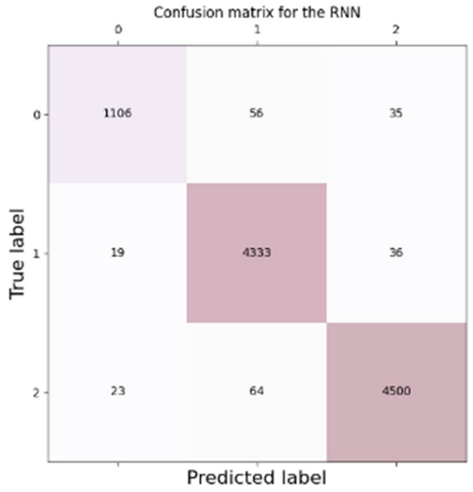

Fig. 6. Confusion Matrix (RNN).

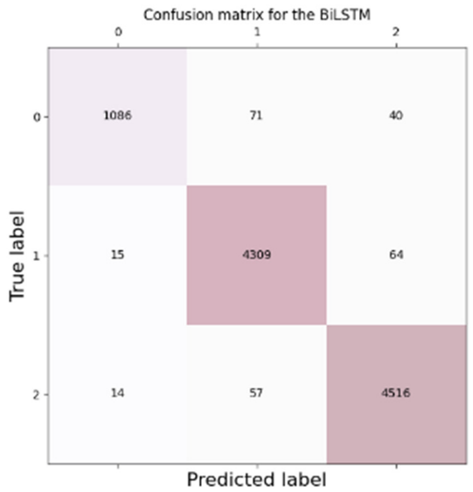

Fig. 7. Confusion Matrix (BiLSTM).

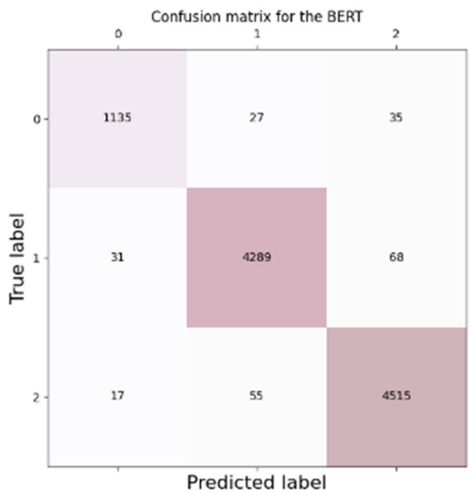

Fig. 8. Confusion Matrix (BERT).

IV. CONCLUSION

Sentiment analysis methods are intended to track and classify emotions expressed in messages submitted on social media platforms. Studies have generally been conducted to classify emotions as good, negative, or neutral. An examination of the sentiments gleaned from Twitter (X) posts was conducted in this study. Two datasets that are readily available in the domain were used. In order to create a well-balanced dataset, we combined the two datasets using concatenation methods. Then, we used the acquired dataset to apply SVM, RNN, bi-directional LSTM, and a freshly developed model, BERT. Based on the discussion in [33], we compared the outcomes using the recall, F1 Score, precision, and accuracy metrics. According to the outcomes, all models delivered accuracy of 96% or higher. With 99% accuracy, the BERT classifier fared better than the others. Future study will entail the application of some more recently developed models and combining them to assess the efficacy of machine learning models in sentiment analysis.

REFERENCES

[1] C. A. Iglesias and A. Moreno, "Sentiment Analysis for Social Media," *Applied Sciences*, vol. 9, no. 23, Jan. 2019, Art. no. 5037, https://doi.org/10.3390/app9235037.

[2] Q. Tul *et al.*, "Sentiment Analysis Using Deep Learning Techniques: A Review," *International Journal of Advanced Computer Science and Applications*, vol. 8, no. 6, pp. 424–433, 2017, https://doi.org/10.14569/IJACSA.2017.080657.

[3] D. Li, R. Rzepka, M. Ptaszynski, and K. Araki, "HEMOS: A novel deep learning-based fine-grained humor detecting method for sentiment analysis of social media," *Information Processing & Management*, vol. 57, no. 6, Nov. 2020, Art. no. 102290, https://doi.org/10.1016/j.ipm.2020.102290.

[4] M. Sinan *et al.*, "Analysis of the mathematical model of cutaneous Leishmaniasis disease," *Alexandria Engineering Journal*, vol. 72, pp. 117–134, Jun. 2023, https://doi.org/10.1016/j.aej.2023.03.065.

[5] P. Savci and B. Das, "Prediction of the customers' interests using sentiment analysis in e-commerce data for comparison of Arabic, English, and Turkish languages," *Journal of King Saud University - Computer and Information Sciences*, vol. 35, no. 3, pp. 227–237, Mar. 2023, https://doi.org/10.1016/j.jksuci.2023.02.017.

[6] M. Rodriguez-Ibanez, A. Casanez-Ventura, F. Castejon-Mateos, and P.-M. Cuenca-Jimenez, "A review on sentiment analysis from social media platforms," *Expert Systems with Applications*, vol. 223, Aug. 2023, Art. no. 119862, https://doi.org/10.1016/j.eswa.2023.119862.

[7] L. Bryan-Smith, J. Godsall, F. George, K. Egode, N. Dethlefs, and D. Parsons, "Real-time social media sentiment analysis for rapid impact assessment of floods," *Computers & Geosciences*, vol. 178, Sep. 2023, Art. no. 105405, https://doi.org/10.1016/j.cageo.2023.105405.

[8] M. Alam, F. Abid, C. Guangpei, and L. V. Yunrong, "Social media sentiment analysis through parallel dilated convolutional neural network for smart city applications," *Computer Communications*, vol. 154, pp. 129–137, Mar. 2020, https://doi.org/10.1016/j.comcom.2020.02.044.

[9] I. Priyadarshini and C. Cotton, "A novel LSTM–CNN–grid search-based deep neural network for sentiment analysis," *The Journal of Supercomputing*, vol. 77, no. 12, pp. 13911–13932, Dec. 2021, https://doi.org/10.1007/s11227-021-03838-w.

[10] A. R. Pathak, M. Pandey, and S. Rautaray, "Topic-level sentiment analysis of social media data using deep learning," *Applied Soft Computing*, vol. 108, Sep. 2021, Art. no. 107440, https://doi.org/10.1016/j.asoc.2021.107440.

[11] Y.-Y. Cheng, Y.-M. Chen, W.-C. Yeh, and Y.-C. Chang, "Valence and Arousal-Infused Bi-Directional LSTM for Sentiment Analysis of Government Social Media Management," *Applied Sciences*, vol. 11, no. 2, Jan. 2021, Art. no. 880, https://doi.org/10.3390/app11020880.

[12] A. Alsayat, "Improving Sentiment Analysis for Social Media Applications Using an Ensemble Deep Learning Language Model," *Arabian Journal for Science and Engineering*, vol. 47, no. 2, pp. 2499–2511, Feb. 2022, https://doi.org/10.1007/s13369-021-06227-w.

[13] P. K. Jain, W. Quamer, V. Saravanan, and R. Pamula, "Employing BERT-DCNN with sentic knowledge base for social media sentiment analysis," *Journal of Ambient Intelligence and Humanized Computing*, vol. 14, no. 8, pp. 10417–10429, Aug. 2023, https://doi.org/10.1007/s12652-022-03698-z.

[14] Z. Jin, X. Zhao, and Y. Liu, "Heterogeneous Graph Network Embedding for Sentiment Analysis on Social Media," *Cognitive Computation*, vol. 13, no. 1, pp. 81–95, Jan. 2021, https://doi.org/10.1007/s12559-020-09793-7.

[15] A. H. Ombabi, W. Ouarda, and A. M. Alimi, "Deep learning CNN–LSTM framework for Arabic sentiment analysis using textual information shared in social networks," *Social Network Analysis and Mining*, vol. 10, no. 1, Jul. 2020, Art. no. 53, https://doi.org/10.1007/s13278-020-00668-1.

[16] M. M. Abdelgwad, T. H. A. Soliman, and A. I. Taloba, "Arabic aspect sentiment polarity classification using BERT," *Journal of Big Data*, vol. 9, no. 1, Dec. 2022, Art. no. 115, https://doi.org/10.1186/s40537-022-00656-6.

[17] M. M.Abdelgwad, T. H. A Soliman, A. I.Taloba, and M. F. Farghaly, "Arabic aspect based sentiment analysis using bidirectional GRU based models," *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 9, pp. 6652–6662, Oct. 2022, https://doi.org/10.1016/j.jksuci.2021.08.030.

[18] W. Alkaberi and F. Assiri, "Predicting the Number of Software Faults using Deep Learning," *Engineering, Technology & Applied Science Research*, vol. 14, no. 2, pp. 13222–13231, Apr. 2024, https://doi.org/10.48084/etasr.6798.

[19] M. Alruily, "Sentiment analysis for predicting stress among workers and classification utilizing CNN: Unveiling the mechanism," *Alexandria Engineering Journal*, vol. 81, pp. 360–370, Oct. 2023, https://doi.org/10.1016/j.aej.2023.09.040.

[20] J. Sangeetha and U. Kumaran, "Sentiment analysis of amazon user reviews using a hybrid approach," *Measurement: Sensors*, vol. 27, Jun. 2023, Art. no. 100790, https://doi.org/10.1016/j.measen.2023.100790.

[21] U. B. Mahadevaswamy and P. Swathi, "Sentiment Analysis using Bidirectional LSTM Network," *Procedia Computer Science*, vol. 218, pp. 45–56, Jan. 2023, https://doi.org/10.1016/j.procs.2022.12.400.

[22] M. Ahmad, S. Aftab, M. Salman, and N. Hameed, "Sentiment Analysis using SVM: A Systematic Literature Review," *International Journal of Advanced Computer Science and Applications*, vol. 9, no. 2, pp. 182–188, Jan. 2018, https://doi.org/10.14569/IJACSA.2018.090226.

[23] H. Batra, N. S. Punn, S. K. Sonbhadra, and S. Agarwal, "BERT-Based Sentiment Analysis: A Software Engineering Perspective," in *32nd International Conference on Database and Expert Systems Applications*, Sep. 2021, pp. 138–148, https://doi.org/10.1007/978-3-030-86472-9_13.

[24] S. Bano and S. Khalid, "BERT-based Extractive Text Summarization of Scholarly Articles: A Novel Architecture," in *International Conference on Artificial Intelligence of Things*, Istanbul, Turkey, Dec. 2022, pp. 1–5, https://doi.org/10.1109/ICAIoT57170.2022.10121826.

[25] S. Bano, S. Khalid, N. M. Tairan, H. Shah, and H. A. Khattak, "Summarization of scholarly articles using BERT and BiGRU: Deep learning-based extractive approach," *Journal of King Saud University - Computer and Information Sciences*, vol. 35, no. 9, Oct. 2023, Art. no. 101739, https://doi.org/10.1016/j.jksuci.2023.101739.

[26] L. Abualigah, M. A. Elaziz, P. Sumari, Z. W. Geem, and A. H. Gandomi, "Reptile Search Algorithm (RSA): A nature-inspired meta-heuristic optimizer," *Expert Systems with Applications*, vol. 191, Apr. 2022, Art. no. 116158, https://doi.org/10.1016/j.eswa.2021.116158.

[27] S. Khalid and S. Wu, "Supporting Scholarly Search by Query Expansion and Citation Analysis," *Engineering, Technology & Applied Science Research*, vol. 10, no. 4, pp. 6102–6108, Aug. 2020, https://doi.org/10.48084/etasr.3655.

[28] S. Khalid, S. Khusro, I. Ullah, and G. Dawson-Amoah, "On The Current State of Scholarly Retrieval Systems," *Engineering, Technology & Applied Science Research*, vol. 9, no. 1, pp. 3863–3870, Feb. 2019, https://doi.org/10.48084/etasr.2448.

[29] M. K. Khan, M. H. Zafar, S. Rashid, M. Mansoor, S. K. R. Moosavi, and F. Sanfilippo, "Improved Reptile Search Optimization Algorithm: Application on Regression and Classification Problems," *Applied Sciences*, vol. 13, no. 2, Jan. 2023, Art. no. 945, https://doi.org/10.3390/app13020945.

[30] H. Alsolai, L. Alsolai, F. N. Al-Wesabi, M. Othman, M. Rizwanullah, and A. A. Abdelmageed, "Automated sign language detection and classification using reptile search algorithm with hybrid deep learning," *Heliyon*, vol. 10, no. 1, Jan. 2024, Art. no. e23252, https://doi.org/10.1016/j.heliyon.2023.e23252.

[31] M. Maashi *et al.*, "Modeling of Reptile Search Algorithm With Deep Learning Approach for Copy Move Image Forgery Detection," *IEEE Access*, vol. 11, pp. 87297–87304, 2023, https://doi.org/10.1109/ACCESS.2023.3304237.

[32] Z. Elgamal, A. Q. M. Sabri, M. Tubishat, D. Tbaishat, S. N. Makhadmeh, and O. A. Alomari, "Improved Reptile Search Optimization Algorithm using Chaotic map and Simulated Annealing for Feature Selection in Medical Filed," *IEEE Access*, vol. 10, pp. 51428–51446, Jan. 2022, https://doi.org/10.1109/access.2022.3174854.

[33] S. Khalid, S. Wu, and F. Zhang, "A multi-objective approach to determining the usefulness of papers in academic search," *Data Technologies and Applications*, vol. 55, no. 5, pp. 734–748, Jan. 2021, https://doi.org/10.1108/DTA-05-2020-0104.

[34] Suran, "Bitcoin Tweets." kaggle, [Online]. Available: https://www.kaggle.com/datasets/skularat/bitcoin-tweets.

[35] F. Gulsen, "Bitcoin Sentiment Analysis." kaggle, [Online]. Available: https://kaggle.com/code/codeblogger/bitcoin-sentiment-analysis.