International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
IJISAE
ISSN:2147-6799 www.ijisae.org Original Research Paper

Meta Heuristic Optimization Algorithm for Twitter Data Sentiment Analysis

L. Sudha Rani*[1], S. Zahoor-Ul-Huq[2], C. Shoba Bindu[3]

Submitted: 17/09/2023 **Revised**: 18/11/2023 **Accepted**: 30/11/2023

Abstract: In recent years, sentiment classification in Twitter using deep learning approaches has gained popularity. Many researchers have focused on Twitter sentiment analysis and have assumed that all words within a tweet have the same polarity, often neglecting the polarity of individual words within the sentence. This paper proposes a novel approach to analyzing tweets, which consists of two main phases: feature selection and classification. In the first phase, the most appropriate features are selected through mutual information analysis. The second phase involves utilizing a Meta Heuristic algorithm to enhance the weights and biases of the multi-layer perceptron network. The study results demonstrate that the MLP network optimized by the Glow-worm Swarm optimization outperforms other existing methods.

Keywords: *Twitter Sentiment Analysis, Optimization, Feature Selection, Multilayer Perception*

1. Introduction

A sentiment analysis is a process utilized to classify various types of views, emotions, and opinions from a text, speech, or tweet into neutral, positive, or negative categories. Twitter's characteristics make it incredibly challenging to perform this process. Due to the length of tweets, they often have misspellings and emoticons. Pre-processing is required before features can be extracted. There are two main types of approaches to perform sentiment analysis: unsupervised learning and supervised learning. The former is focused on learning from sentiment lexicons, while the latter is on classification [1]. The goal of training networks is to find the optimal set of connections and biases to minimize the error rate when it comes to approximation or classification. Gradient-based methods are commonly used in this process [2]. One of the most common methods used for training networks is the back-propagation algorithm. For most complex problems, the use of gradient-based methods is not ideal. They tend to have high dependence on the initial solution [3-4] and are prone to failure when it comes to convergence [5].

According to previous studies, sentiment classifiers are subject-dependent, and they can only perform well on certain concepts [6-7]. There is no single best classifier for every concept. Instead, there are multiple classifiers that can perform well on various concepts. This paper aims to investigate the various advantages of using meta-heuristic methods such as the glowworm swarm optimization, Genetic Algorithm, and biogeography-based optimization to improve the performance of sentiment analysis.

The rest of the paper is organized as follows: Section 2 deals with the recent surveys on Meta heuristic algorithms in the field of twitter sentiment analysis. Section 3 explains about the proposed method for sentiment analysis. Section 4 explains about the experimental analysis on the proposed and existing twitter sentiment analysis. Finally, section 5 concludes the research work.

2. Literature Survey

Studies on the application of meta-heuristic algorithms for sentiment analysis on Twitter can be categorized into two groups: (i) selecting the optimal subset of extracted features, and (ii) optimizing the classifier.

2.1. Feature Selection using Meta Heuristic Algorithms

The authors in [8] used a combination of GA and CRF to classify sentiments. The results of the experiment showed that the proposed algorithm performed well in real-world applications. The authors in [9] performed a feature selection procedure using the binary shuffle frog algorithm and a machine learning framework. The pre-processing phase involved the use of stop words and stems. The features were then extracted using the TF-IDF. The authors then used the proposed algorithm to classify tweets into positive or negative. They did so by using the KNN, naive Bayes, LMT, and RBF network classifiers. The authors evaluated the proposed algorithm against a corpus of tweets from Stanford University. The corpus included 875 negative and 325 positive tweets. The results of the experiment revealed that the RBF network classifier performed better than the KNN, LMT, and NB [20].

[1] Research Scholar, Dept.of CSE, JNTUA Anantapur, Andhra Pradesh, India, Asst.Professor G. Pulla Reddy Engineering College, Kurnool, Affiliated toJNTU Anantapur, ORCID ID : 0000-0001-5505-1524
[2] Professor, Dept. of CSE, G. Pulla Reddy Engineering College, Kurnool, AP, India
[3] Professor, Dept.of CSE, JNTUA Anantapur, Andhra Pradesh, India

2.2. Classifier Optimization using Meta Heuristic Algorithms

In [10], the authors presented a method that utilizes the support vector machine (SVM) and particle swarm optimization (PSO) to classify Twitter movie reviews into two categories: watchable and non-watchable. The PSO framework was employed to optimize the parameters of the SVM, and the method was evaluated accordingly. The results showed that the classification accuracy increased from 71.87% to 77% after PSO optimization. The study also found that the unigram algorithm outperformed the bigram and trigram extraction methods in terms of feature selection [19]. However, the authors noted the need to include the neutral class in the sentiment classification.

In [11], a method for analyzing Twitter sentiment was introduced using cuckoo search (CS) and K-means. The proposed method aimed to enhance the performance of CS by modifying its random initialization process. The study compared the results of the proposed method with those of other algorithms and demonstrated its superior efficiency. Four Twitter datasets were analyzed using different metaheuristic algorithms, and the proposed method showed efficient classification. However, the study did not utilize any feature selection techniques to improve accuracy [16-18].

This paper presents a feature selection method based on mutual information, followed by the application of meta-heuristic algorithms to improve classifier performance. The authors classified tweets into three categories: positive, negative, and irrelevant.

3. Meta Heuristic Approach for Sentiment Analysis

The proposed approach is divided into two phases: feature selection and classification model development. The feature selection stage aims to identify the features that will assist the classification model in achieving optimal performance during the second stage. The first stage employs a mutual information technique to identify potential features. The second stage utilizes meta-heuristic algorithms to train the algorithm on the selected subsets from the previous stage.

The proposed approach is composed of four phases: pre-processing of tweets, feature extraction, feature selection, and classification of tweets utilizing the hybrid algorithm of MLP. The system architecture of the proposed approach is illustrated in Figure 1.

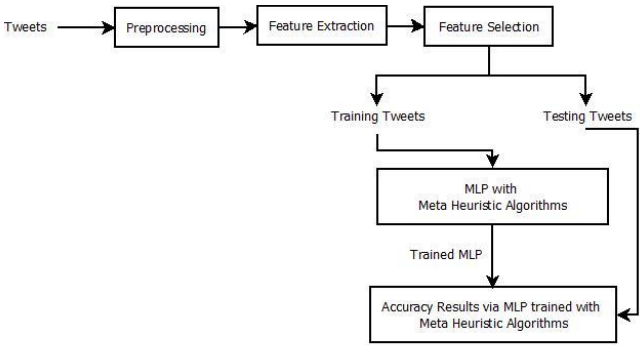

Fig. 1. System Architecture for Twitter Sentiment Analysis.

3.1. Pre-processing

To extract useful features, the tweets in the dataset need to be pre-processed to remove unwanted words, URLs, stop words, and other noise. The pre-processing step is crucial as it determines the effectiveness of subsequent steps. Its goal is to enhance the machine readability of the data and reduce ambiguity. In this study, noise data was removed from the dataset using the following steps:

The cleaning process aims to remove unwanted data from the dataset to make it more machine-readable. The following steps are usually taken to clean the data:

i. URLs in tweets usually do not provide any useful information and can be removed.

ii. While hash-tags provide useful information, removing the # symbol makes it easier for machines to read them.

iii. Parentheses, forward slash, backward slash, and dash can be removed as they do not add any valuable information.

iv. Punctuation marks and digits can be removed as they do not add any significant value to the text.

v. Multiple white spaces can be replaced with a single white space to make the text more compact.

vi. Converting all the words into lower case makes it easier for the machine to process them.

vii.Stop words such as "a," "is," "the," etc., are commonly used in tweets and do not add any significant information. Therefore, they can be removed from the dataset.

It's important to note that tokenization and stemming are separate processes. Tokenization breaks text into individual words or n-grams, while stemming reduces words to their root form. The purpose of stemming is to group together different forms of the same word so that they can be treated as one word. For example, "running" and "runs" would both be stemmed to "run". This can be useful in text analysis tasks such as sentiment analysis where the sentiment of related words should be treated equally.

3.2. Feature Extraction

To expand on the explanation of feature extraction, the process of transforming text data into numerical features involves converting each tweet into a vector of numerical values that can be used for machine learning algorithms. In the case of terms presence and frequency (TP-F) feature extraction; this involves counting the frequency of each individual word or n-gram in each tweet and creating a feature vector where each dimension represents the frequency count of a particular word or n-gram in a tweet. This process essentially converts text data into a numerical representation that can be used for machine learning algorithms. The resulting feature vectors can then be used as input to train a classification model to predict sentiment labels for new tweets.

3.3 Feature Selection

In the feature extraction step, many features is usually generated, especially in the case of sentiment classification. Due to the complexity of training a large set of classifiers, it is often expensive to train them. With the help of feature selection techniques, it can reduce the computational costs and improve the performance of the classification. In this paper, we introduce the MI technique, which is a type of feature selection method that considers a mathematical equation. It then selects a set of features that are then used with a classifier.

3.4 Classifier

In this section, the process of using three different meta-heuristic algorithms - GA, BBO, and GSO - as trainers for the MLP network is described. The first step in this process is to obtain an initial solution, which is achieved by setting up the MLP network. After that, the meta-heuristic algorithms are used to minimize the classification biases and weights, which are the key factors in achieving high classification accuracy.

3.4.1 Multi-Layer Perception

MLP stands for Multi-Layer Perceptron, and it is a type of neural network that consists of multiple layers of interconnected neurons, arranged in a hierarchical fashion. The first layer is an input layer, and the last layer is an output layer. The hidden layers are used to produce the network's outputs. Multi-layer perceptron network composed of multiple layers of linked neurons. The connections between these layers are referred to as weights W, where W is defined with 0 and 1 and their output value is computed in two phases.

In the first phase, the summation of the weighted inputs are computing using the Eq. 1

$$h_k = \sum_{i=1}^{n} W_{ik}^{H} U_i + \beta_k^{H}, \forall(k) \in \{1,2,\ldots,l\}$$

(1)

Where the input variable is represented with U_i, n represents the total inputs, W_{ik}^{H} represents the weights in between the input neuron i and the hidden neuron k, the k^{th} hidden neuron bias is represented with β_k^{H}.

In the second phase, the hidden layer neuron output value is determined by computing a weighted sum of its inputs and applying an activation function to the result. The activation function, which is often the sigmoid function, maps the resulting value to a specific range and helps the network learn complex patterns by introducing nonlinearity.

$$h_k = sig(h_k) = \frac{1}{1+e^{-h_k}}, \forall(k) \in \{1,2,\ldots,l\}$$

(2)

The network final output is computed using the Eq. 3

$$o_x = \sum_{k=1}^{m} W_{xk}^{O} H_k + \beta_x^{O}, \forall(x) \in \{1,2,\ldots,j\},$$

(3)

$$o_x = sig(o_x) = \frac{1}{1+e^{-h_x}}, \forall(x) \in \{1,2,\ldots,j\},$$

(4)

Where W_{xk}^{O} represents the weights in between the hidden neuron k and the hidden output neuron x, the x^h hidden neuron bias is represented with β_x^{O}.

The proposed approach was developed taking into account the encoding scheme of search agents and the fitness of the algorithms.

Encoding Scheme: The candidates in the GSO algorithm is encoded with real number vectors with in the [0, 1] range. The three components of these vectors include the connection weights of the input layer to the hidden layer and hidden layer to the output layer. The problem dimension G is computed using the following Equation 5.

$$G = (n*m) + m + (m*j) + j$$

(5)

Where n denotes the input variables, m denotes the neurons in the hidden layer and j denotes the neurons in the output layer.

Fitness Function:

The fitness of each individual is determined by evaluating its performance in the MLP network. The individual's vector of weights and biases is passed to the MLP, and the mean squared error (MSE) criterion is computed based on the difference between the predicted and actual values of all the training instances. The process is repeated for a maximum number of iterations until an optimal solution is obtained, which corresponds to the optimal weights and biases for the MLP network. Eq. 6 shows the computation of the MSE.

$$MSE = \sum_{l=1}^{L} \frac{\sum_{x=1}^{j} (o_x^l - d_x^l)^2}{L} \tag{6}$$

Where L represents the training data set instances, j denotes the total outputs, d_x^l denotes the actual output of x^{th} input and o_x^l denotes the predicted output of x^{th} input.

4. Glow-Worm Swarm Optimization Algorithm

The GSO algorithm is based on the glow-worm's behaviour. When a glow-worm produces more light, it is closer to its actual position, and it has a high objective function. The GSO algorithm can be described in four phases, namely initialization, luciferin updating, moving, and local radial range updating. The algorithm can be formulated using the steps outlined in Algorithm 1 [12] [13]. Initially, the glow-worms are placed randomly in the search space, and each glow-worm is assigned an equal amount of luciferin. The objective function for each glowworm δ is given as $f(y_\delta(l+1))$ at the current location $y_\delta(l)$. The luciferin $g_\delta(l+1)$ value is given as follows.

$$g_\delta(l+1) = (1-p)g_\delta(l) + \gamma f(y_\delta(l+1)) \tag{7}$$

Where $g_\delta(l)$ represents the glowworm luciferin value δ at time t. The luciferin decay coefficient is represented with P (0<p<1) and the luciferin enhancement coefficient is represented with γ.

In the next stage, the glowworm moves towards their neighbouring glowworms Q which is having the highest luciferin value within the radial range of γ_d.

$$Q_\delta(l) = \{q : \| y_q(l) - y_\delta(l) \| \leq \gamma_d^\delta ; g_\delta < g_q(l)\} \tag{8}$$

Where the neighbouring set is represented with $Q_\delta(l)$, q represents the glowworm index which is nearer to the δ, $y_q(l)$ and $y_\delta(l)$ denotes the luciferin values of q and δ. γ_d^δ represents the radial range. ||y|| represents the Euclidian norm of y.

$$\Pr_{\delta q}(l) = \frac{g_q(l) - g_\delta(l)}{\sum_{s \in Q_\delta(l)} g_w(l) - g_\delta(l)} \tag{9}$$

Where $Pr_{\delta q}$ represents the probability of glowworm δ moving towards the neighbouring glowworm q.

$$y_\delta(l+1) = y_\delta(l) + s\left(\frac{y_q(l) - y_\delta(l)}{\| y_q(l) - y_\delta(l) \|} \right) \tag{10}$$

Where the glowworm current and new location is given as $y_\delta(l+1)$ and $y_\delta(l)$. s represents the moving step size of the glowworm.

At last, The neighbouring set is formulated by updating the local radial range γ_d^δ.

$$\gamma_d^\delta(l+1) = \min\{\gamma_s, \max\{0, \gamma_d^\delta(l) + \beta(q_l - |Q_\delta(l)|)\}\} \tag{11}$$

Where β represents the changing rate of the radial range of neighbourhood gloworm.

Algorithm 1: GSO Algorithm

Step 1: Initialize P, s, β, g_l, ∀δ

Step 2: Set $g_\delta(0) = g_0$ ∀δ

Step 3: Set $\gamma_d^\delta(0) = \gamma_0$ while the termination condition is not satisfied do

Step 4: for δ ∈ n do

Step 5

$$g_\delta(l+1) = (1-p)g_\delta(l) + \gamma f(y_\delta(l+1))$$

Step 6: $Q_\delta(l) = \{q : \| y_q(l) - y_\delta(l) \| \leq \gamma_d^\delta ; g_\delta < g_q(l)\}$

Step 7: for each $q \in Q_\delta(l)$ *do*

Step 8:

$$\Pr_{\delta q}(l) = \frac{g_q(l) - g_\delta(l)}{\sum_{s \in Q_\delta(l)} g_w(l) - g_\delta(l)}$$

Step 9: $y_\delta(l+1) = y_\delta(l) + s\left(\frac{y_q(l) - y_\delta(l)}{\| y_q(l) - y_\delta(l) \|} \right)$

Step 10:
$\gamma_d^\delta(l+1) = \min\{\gamma_s, \max\{0, \gamma_d^\delta(l) + \beta(q_l - |Q_\delta(l)|)\}\}$

Step 11: $l=l+1$

Step 12: return Optimal Solution

5. Experimental Analysis

5.1. Types of Graphics

The proposed method is trained on the supervised tweets collected from the twitter streaming API [14]. The tweets contain the positive (e.g. #joy and #happy) and negative tweets (e.g. #angry, #sadness, #frustrated). The filtering is employed to remove the quotes, retweets, duplicates, tweets and spams tweeted in other than the English language. To solve the issue in [15], the tweets with # tags in the middle of the text are considered as noisy tweets. The proposed model is evaluated based on the two settings such as balanced and unbalanced. The balanced settings include the equal number of positive tweets and negative tweets. The unbalanced settings include the 2:1 ratio (i.e. 2 portions of positive tweets and 1 portion of negative tweets). The 84% of the tweets are topic based and author based contexts whereas small portion of tweets are occupied by the conversion based context. Table 1 shows the description of the dataset.

The performance of the word embedding model is tested with the 10 fold cross validation. The complete twitter data set is divided into 10 equal portions. Each portion is cracked by the proposed model which is trained by nine other portions. One portion is selected randomly from the nine portions are used as the dataset:

Table 1: Twitter corpus [14]

Domain	Number of Tweets	Conversation based context	Author based context	Topic based context
Positive tweets	5855	289	4753	5345
Negative tweets	5855	483	4748	5358

The second phase of the proposed approach involves the use of a meta-heuristic algorithm on the testing data to improve the MLP classifier. The fitness function and accuracy rate are then calculated and evaluated. Table 2 shows the parameter settings for the Meta heuristic algorithms.

Table 2: Glow-worm Swarm Optimization Parameter Settings

Parameter	Value
No. of neighbours	6
Neighbourhood radius range	0.08
Initial value of Luciferin	0.05
Luciferin enhancement Coefficient(γ)	0.6
Luciferin decay Coefficient(P)	0.4
Moving step size	0.03

The fitness function value and classification accuracy rate were used as two parameters to evaluate the proposed method, which was executed ten times along with an existing algorithm. The results of the evaluation, including the average, standard deviation, and best values of the fitness function and classification accuracy rate, are presented in Table 3 and Table 4.

Table 3: Classification Accuracy of the Proposed and Existing Algorithms

Dataset		Proposed GSOMLP	BBOMLP	GAMLP
Twitter Corpus	Average	69.00%	49.00%	54.80%
	Standard Deviation	0.0078	0.0125	0
	Best	58.62%	52.49%	54.30%

Table 4: Fitness Value of the Proposed and Existing Algorithms

Dataset		Proposed GSOMLP	BBOMLP	GAMLP
Twitter Corpus	Average	0.5427	0.5015	0.4987
	Standard Deviation	0.0071	0.0067	0.0072
	Best	0.5148	0.4975	0.4812

Table 3 presents the accuracy rates of the three classification algorithms when applied to the Twitter dataset. The highest accuracy rate of 69% was achieved by the GSO-MLP algorithm, while the lowest rate of 49% was obtained by BBO. On the other hand, Table 4 displays the average, best, and standard deviation of the mean squared error (MSE) values for the three algorithms. Although the GSO-MLP had the highest MSE value, it was found to be like the other two algorithms. Furthermore, Figure 2 depicts the convergence behavior of the different methods, including the proposed method, through a plot that shows the number of iterations and the average MSE value over ten runs. The convergence plot indicates the faster convergence of the proposed method compared to the other methods.

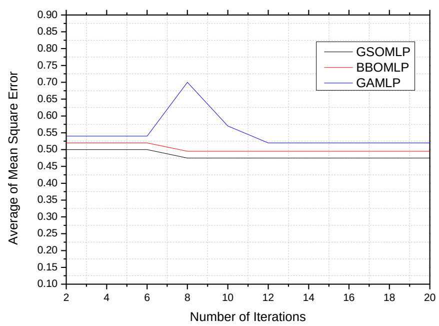

Fig. 2. Convergence of Proposed and Existing Algorithms

Figures 3 and 4 display the different algorithms utilized for training networks on the Twitter dataset using GSO, GA, and BBO. Figure 3 presents plots that analyze the variability in the mean MSE values obtained by each trainer during the last iteration, based on ten MSEs. Conversely, Figure 4 comprises plots that illustrate the variability in the accuracy rates of the classification across different ranges of values in the Twitter dataset. Overall, the GSO algorithm proved to be the most effective for training networks on the Twitter dataset.

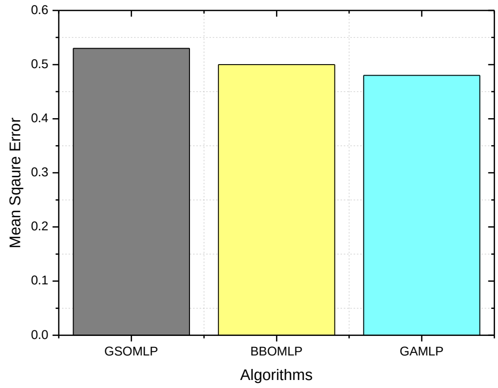

Fig. 3. MSE of Proposed and Existing Algorithms

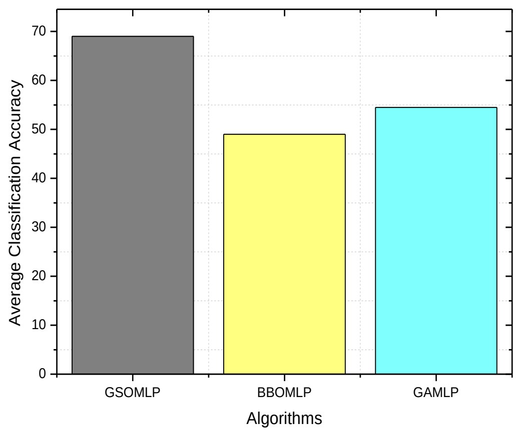

Fig. 4. Average Classification Accuracy of the Proposed and Existing Algorithms

6. Conclusion

This paper proposes a hybrid approach to analyze Twitter sentiment. It is performed in two phases: the first one is for feature selection, while the second one is for classification. The first stage involves the use of MI as a feature selection method, while the second stage involves the use of hybrid and MLP techniques. The results of the evaluation revealed that GSOMLP outperforms GAMLP and BBOMLP in terms of its ability to classify tweets. As a result, further studies on Twitter datasets are required to improve the accuracy rates and speed up the process of classification.

7. References & Footnotes

Author contributions

Sudha Rani L : Conceptualization, Methodology, Software, Field study **Zahoor-Ul-Huq S:** Data curation, Writing-Original draft preparation, Software, Validation., Field study **Shoba Bindu C:** Visualization, Investigation, Writing-Reviewing and Editing.

Conflicts of interest

The authors declare no conflicts of interest.

References

References should be written like this. Please do not use any web-site [URL] as a citation. Do not put an empty line between references.

[1] A. C. Pandey, D. S. Rajpoot, and M. Saraswat, "Twitter sentiment analysis using hybrid cuckoo search method," *Information Processing & Management*, vol. 53, no. 4, pp. 764–779, 2017.

[2] R. Obiedat, O. Harfoushi, R. Qaddoura, L. Al-Qaisi, and A. M. Al-Zoubi, "An evolutionary-based sentiment analysis approach for enhancing government decisions during COVID-19 pandemic: The case of Jordan," *Applied Sciences*, vol. 11, no. 19, p. 9080, 2021.

[3] N. Yuvaraj and A. Sabari, "Twitter sentiment classification using binary shuffled frog algorithm," *Intelligent Automation & Soft Computing*, vol. 23, no. 2, pp. 373–381, 2017.

[4] M. A. Hassonah, R. Al-Sayyed, A. Rodan, A. M. Al-Zoubi, I. Aljarah, and H. Faris, "An efficient hybrid filter and evolutionary wrapper approach for sentiment analysis of various topics on Twitter," *Knowledge-Based Systems*, vol. 192, p. 105353, 2020.

[5] A. Kumar and A. Jaiswal, "Swarm intelligence based optimal feature selection for enhanced predictive sentiment accuracy on Twitter," *Multimedia Tools and Applications*, vol. 78, pp. 29529-29553, 2019.

[6] J. Kaur, S. S. Sehra, and S. K. Sehra, "Sentiment analysis of twitter data using hybrid method of support vector machine and ant colony optimization," *International Journal of Computer Science and Information Security (IJCSIS)*, vol. 14, no. 7, 2016.

[7] M. I. A. Latiffi, M. R. Yaakub, and I. S. Ahmad, "Flower Pollination Algorithm for Feature Selection in Tweets Sentiment Analysis," *International Journal of Advanced Computer Science and Applications*, vol. 13, no. 5, 2022.

[8] J. Zhu, H. Wang, and J. Mao, "Sentiment classification using genetic algorithm and conditional random

fields," in *Information Management and Engineering*, The 2nd IEEE Int. Conf. on, pp. 193–196, IEEE, 2010.

[9] N. Yuvaraj and A. Sabari, "Twitter sentiment classification using binary shuffled frog algorithm," *Intelligent Automation & Soft Computing*, vol. 23, no. 2, pp. 373–381, 2017.

[10] A. S. H. Basari, B. Hussin, I. G. P. Ananta, and J. Zeniarja, "Opinion mining of movie review using hybrid method of support vector machine and particle swarm optimization," *Procedia Engineering*, vol. 53, pp. 453–462, 2013.

[11] A. C. Pandey, D. S. Rajpoot, and M. Saraswat, "Twitter sentiment analysis using hybrid cuckoo search method," *Information Processing & Management*, vol. 53, no. 4, pp. 764–779, 2017.

[12] D. A. Alboaneen, H. Tianfield, and Y. Zhang, "Glowworm swarm optimisation algorithm for virtual machine placement in cloud computing," in *Ubiquitous Intelligence & Comp., Advanced and Trusted Comp., Scalable Comp. and Communications, Cloud and Big Data Comp., Internet of People, and Smart World Congress (UIC/ATC/ScalCom/CBDCom/IoP/SmartWorld), Intl IEEE Conf.*, pp. 808–814, IEEE, 2016.

[13] D. Alboaneen, H. Tianfield, and Y. Zhang, "Glowworm swarm optimisation based task scheduling for cloud computing," in *Proc. of the Int. Conf. on Internet of Things and Cloud Comp*., (Cambridge, 22-23 Mar), pp. 1–7, ACM, 2017.

[14] http://dev.twitter.com/docs/streaming-apis accessed on 09/12/2019.

[15] Davidov, D.; Tsur, O.; and Rappoport, A. 2010. Enhanced sentiment learning using twitter hashtags and smileys. In *Proceedings of the 23rd International Conference on Computational Linguistics* (COLING), 241–249.

[16] U. Can and B. Alatas, "A novel approach for efficient stance detection in online social networks with metaheuristic optimization," *Technology in Society*, vol. 64, p. 101501, 2021.

[17] M. Tenemaza, S. Lujan-Mora, A. De Antonio, and J. Ramirez, "Improving itinerary recommendations for tourists through metaheuristic algorithms: an optimization proposal," *IEEE Access*, vol. 8, pp. 79003-79023, 2020.

[18] A. Jain, B. P. Nandi, C. Gupta, and D. K. Tayal, "Senti-NSetPSO: large-sized document-level sentiment analysis using Neutrosophic Set and particle swarm optimization," *Soft Computing*, vol. 24, no. 1, pp. 3-15, 2020.

[19] D. Nagaraju and V. Saritha, "An evolutionary multi-objective approach for resource scheduling in mobile cloud computing," *Int J Intell Eng Syst*, vol. 10, no. 1, pp. 12-21, 2017.

[20] H. Naz, S. Ahuja, and D. Kumar, "DT-FNN based effective hybrid classification scheme for twitter sentiment analysis," *Multimedia Tools and Applications*, vol. 80, pp. 11443-11458, 2021.