TWITTER POLARITY CLASSIFICATION MODEL USING HYBRID GENETIC ALGORITHM AND CUCKOO SEARCH OPTIMIZATION ALGORITHM

Priyanka[1*], Kirti Walia[2]

Abstract

Nowadays, Twitter sentiment analysis has become one of the most fascinating academic disciplines. It combines methods for natural language processing with data analysis methodologies for developing such systems. This research presented an efficient approach for analysing Twitter sentiments. A Hybrid Optimization Model using Steady State Genetic Algorithm (HOM-SSGA) for Twitter Polarity Analysis is designed in this article. The suggested approach used a machine learning algorithm to identify the positive and negative polarity of tweets. During the training stage, the proposed system represented the input-labelled tweets using various approaches and feature sets. In this study, the suggested system includes three primary phases: data gathering, preparation, and sentiment categorization from precompiled tweets. The Twitter-Sanders-Apple 2 database was used for preliminary sentiment analysis on Twitter. On average, the raw data gathered tweets have more noise in the form of positive emojis, punctuation marks, and negative emojis, which were removed to get a more accurate result. The work employs a steady-state genetic algorithm for attribute selection and a cuckoo search for extracting features. The experimental findings and statistical assessment confirm that the suggested technique surpasses the current methods such as Artificial Neural Network (ANN), Bidirectional Encoding Representation from Transformation (BERT), Term Frequency-Inverse Document Frequency (TF-IDF), and Feed Forward Back Propagation Neural Network (FFBPNN). The suggested technique has improved effects for the analysis of social network and media data in future studies.

Keywords – Classification, Genetic Algorithm, Cuckoo Search Algorithm, Social Media Analysis

[1*,2] Department of Computer Applications, Chandigarh University Gharuan Punjab, India.

***Corresponding Author :-** Priyanka
*Department of Computer Applications, Chandigarh University Gharuan Punjab, India.
Email:- priyankatuli1986@gmail.com,9501489236

DOI: - 10.48047/ecb/2023.12.si5a.0160

1. Introduction to social media sentiment analysis

Messaging, social media connections, blogging, and tweeting are now the most popular online activities. Twitter is one of the most famous microblogging platforms and might be regarded as one of the biggest user-generated data sites with a vast quantity of organised and unstructured information [1]. According to the interests of the viewers, the uploaded tweets might indicate their thoughts on various issues and their polarity regarding these subjects.

Social media has evolved into a legitimate means of communication for individuals to express their opinions and points of view on any topic [2]. Numerous studies must investigate these consumer viewpoints. In addition, they depend on the comments offered by diverse Internet users. This may greatly alter the product's purchasing behaviour. Consequently, studying the views or feelings of the user arises as a crucial area of research. Sentiment analysis (SA) is the study of recognising and classifying the public's views, sentiments, emotions, and attitudes about any subject, person, or event. Additionally, the view is classified as good, negative, or neutral [3].

Sentiment classification combines data mining, text analytics, and web usage mining methods to discover, extract, and identify views, feelings, and sentiments regarding certain subjects [4]. It may be used with many data sources, including reviews, blogs, and news. In several application fields, such as analysing marketing efforts, rating movies, and measuring customer happiness, identifying people's views provides very relevant information.

In sentiment classification, the primary data consists of the online text shared by social media members. Twitter, one of these social media platforms, has become the dominant source for online data interchange, giving a wide platform for sentiment research [5]. Twitter is an extremely popular social networking platform that enables authorised members to submit 140-character tweets or brief comments. Twitter's dataset is among the biggest, with 200 million members posting 400 million messages or tweets every day [6]. Twitter members often offer their own opinions on a variety of topics, such as their approval or disapproval of politicians and their views on goods, as well as current affairs and family-related occurrences. Furthermore, people reduce the number of letters in tweets by adding abbreviations and symbols like emojis [7]. Consequently, the examination of these tweets may be utilised to identify strong opinions and feelings about any given issue. Various individuals have previously used Twitter data to forecast financial markets, box office revenues for movies, identify customers with negative feelings, and so on [8]. The primary objective of sentiment assessment is to discover how people feel about a certain issue. This research provides a unique classification technique for sentiment assessment on the Twitter database [9–10].

Twitter's sentiment assessment differs from that of other social media systems in several ways: (i) consumers tend to use very brief tweets to convey their feelings and conditions; (ii) consumers may use notations and emojis to conserve characters; and (iii) numerous linguistic recognition difficulties arise from design problems. This article investigated several methods of selected features that may be used to effectively represent Twitter posts. As demonstrated by most text mining algorithms, the retrieved features might result in a complicated computing issue owing to the enormous size of the created feature space. Feature selection approaches such as data gains, mutual data, etc., as well as feature conversion methods such as feature hashing and evaluation, might be used to address such a challenge.

The primary contribution of the article is listed below:

- A detailed analysis of the existing literature related to social media sentiment analysis is done to understand the issues in the domain.
- The hybrid Optimization Model for Twitter Polarity Analysis (HOM-TPA) is designed in this research to detect the polarity of the user's Twitter posts.
- The genetic algorithm is used for feature selection and the Cuckoo Search (CS) algorithm is used to extract features.
- The suggested method is compared with different classification methods like NB, SVM and FFBPNN.

The parts that follow are grouped as shown below. The second section describes the history of social media sentiment analysis and its effects. In this section 3, the suggested Hybrid Optimization Model for Twitter Polarity Analysis (HOM-TPA) is constructed using GA and CS algorithms. Section 4 illustrates the experimental results and conclusions of the Twitter sentiment analysis. Section 5 summarises the suggested HOM-TPA method's conclusion and results.

2. Background to the Twitter sentimental analysis and its outcomes

Scholars have created several approaches for Twitter sentiment classification. In this section, a quick appraisal of a handful of important additions to the current literature is provided.

This research concentrates on how to combine text data from Twitter messages with characteristics of emotion diffusion to improve the effectiveness of sentiment classification on Twitter information [11]. The research first investigates a phenomenon known as "sentiment reversing" and identifies several fascinating features of sentiment analysis. The research explores the interconnections between the text data of Twitter posts and trends of emotion dispersion, and the research presents an iterative technique for predicting the sentiment polarity conveyed in Twitter posts.

This article presents a transformer-based technique for sentiment classification that encrypts depiction from a transmitter and relates deep learning based experiential encoding to improve the quality of Twitter posts by deleting noise and taking into consideration word feelings, polysemy, syntax, and semantic information [12]. A multimodal network of long-term and short-term memories is also used in the research to figure out how someone feels about a message.

This paper intends to offer a novel, two-step method for Twitter sentiment classification [13]. Initially, the jargon used in tweets, including emojis and symbols, is converted into plain text using language-independent or readily applicable processes. The resultant tweets are then classified using the language system, which was educated on plain text rather than tweets for two purposes: (1) pre-trained systems on plain text are readily accessible in a variety of formats, eliminating resource- and time-intensive model directly on tweets from the start; (2) accessible plain text datasets are bigger than tweet-only corpora, enabling better effectiveness.

This research intends to do a comprehensive sentiment classification of tweets using methods from machine learning [14] and ordinal modeling. The suggested technique involves first filtering tweets and then using an effective feature extraction algorithm. Furthermore, grading and balancing are performed on these characteristics under several classifications. Experimental results indicate that the suggested approach can accurately identify ordinal regression using machine learning techniques.

Sentiment classification in social media is a basic topic with several fascinating applications. Most of the existing social media sentiment classification algorithms evaluate the polarity of sentiment based solely on text data and disregard additional data on these sites. This study presents a neural network model that combines information on user activity inside a particular text (tweet) [15]. Convolutional Neural Networks (CNN) are used in this article. The suggested model surpasses existing baseline methods, demonstrating that going beyond the contents of a communication (a tweet) is advantageous for sentiment analysis since it gives the classifiers a thorough grasp of the job.

This study obtained data from the tweeting platform (Twitter) on agricultural protests to comprehend the worldwide public's views [16]. The research categorised and analysed the attitudes of around 20,000 tweets about the demonstration using algorithms. Using Bag of Words and Term Frequency-Inverse Document Frequency (TF-IDF), the scholars did the investigation and observed that Bag of Words functioned better than TF-IDF.

They suggest a quantum-inspired emotion reconstruction framework. Not only does this approach reflect the semantic information of texts, but it can also collect sentiment data [17]. Since words and adjectives are strong markers of subjective emotion, this approach starts by extracting sentiments that fit the predefined sentiment categories based on adjectival and adverbial phrases. Furthermore, both individual words and emotion phrases are represented as a group of projections, which are eventually contained in density matrices using maximum possibility prediction. These density vectors effectively incorporate sentiment data into the textual reconstructions.

In this paper, a four-phase paradigm for Twitter sentiment assessment [18] is proposed. The encoder for producing sentence descriptions is founded on the Bidirectional Encoding Representation from Transformation (BERT) model. For more efficient use of this approach, several categorization models are used. In addition, the research combines pre-trained word embedding models with the BERT representation approach for improving sentiment categorization. The empirical findings demonstrate superior execution as

compared to the baseline architecture across all databases.

This multimodal text mixes words and pictures to form a distinct visual vocabulary that must be studied since it can change, affirm, or evaluate the sentiment's polarization. The research provides a multimodal emotion assessment methodology to estimate the sentiment polarisation and score for every inbound tweet, i.e., textual, picture, infographic, and typographical [19]. Sentiment analysis of images is performed using SentiBank and SentiStrength scores for areas in conjunction with convolutional neural networks. Using a unique context-aware hybridization (vocabulary and artificial learning) method, the sentiment of the text is determined. Multimodal emotion rating is accomplished by isolating text from pictures using an optical mark recognition system and then averaging the separately computed scores for the image and phrase.

The objective of the work presented in this study is to identify and evaluate the mood and emotion displayed by individuals in their Twitter tweets and to utilise this information to generate suggestions [20]. The research gathered tweets and answers on a handful of issues and compiled a dataset including text, client, emotions, attitude, etc. The research utilised the database to identify sentiment and emotions from tweets and their responses and to calculate the influence ratings of users based on a variety of user- and tweet-based criteria. The research then utilised this information to develop broad and individualised suggestions for Twitter customers depending on their activities.

A lot of studies have done sentiment assessment using various mechanisms and on various kinds of databases, but only a small number of studies have used optimising approaches for sentiment classification. Therefore, the blended technique for feature extraction and sentiment analysis is used in this suggested study. Different classifiers have also been employed to increase the accuracy of sentiment classification. However, there is room for the use of several classifications.

3. Proposed Hybrid Optimization Model for Twitter Polarity Analysis

This section will briefly outline the elements of the suggested Twitter sentiment assessment method. As seen in Fig. 1, the suggested scheme operates in two stages: training and categorization. The training stage's goal is to create a classifier model that can distinguish between positive and negative tweets based on labelled tweet input datasets. In the categorization step, the training categorization model will identify unmarked tweets as positive or negative. Four processes comprise the scheme: preprocessing, feature extraction, component selection, and a classification algorithm for sentiment assessment.

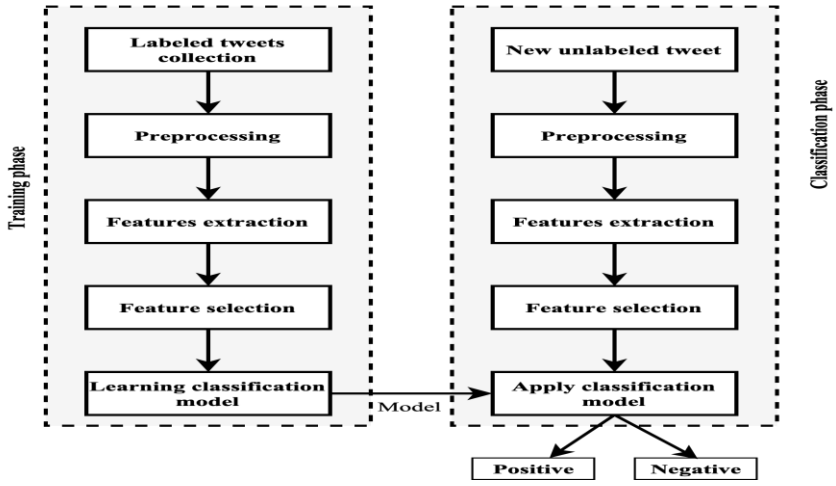

Fig. 1. The architecture of the proposed HOM-TPA system

3.1 Twitter data collection

Initial Twitter sentiment information is acquired from the database (twitter-sanders-apple 2),

including 1000 tweets, and is loaded straight into the Light side [21]. There are 500 tweets with negative emotions and 500 tweets with positive emotions out of a total of 1000 tweets. After getting information from Twitter, the raw tweets are tokenized so that hashtags, sentences, and important words can be pulled out.

3.2 Tokenization

The tokenization approach is used to divide tweets into signals, hashtags, or significant phrases, which are widely developed for parsing and text analytics [22]. Initialize tokenization by segmenting the tweets into terms and locating the segregated words' borders. Typically, a work boundary begins with one phrase and concludes with another. Additionally, tokenization is straightforward if the divided tweets have more spacing between them. When tokenization happens, punctuation marks are regarded as white spaces if there are no existing white spaces. In this section, the token conceptions are specified before any computation.

3.3 Preprocessing

This stage's primary purpose is to employ natural language processing methods to analyse the input tweet and make it acceptable for the subsequent step of accurately extracting features.

Encoding, text cleaning, point-of-sale Labeling and stemming are the sub-steps of the preparation phase. The preprocessing began by tokenizing (separating) the input text into individual words (called tokens). Each token may symbolise a widely occurring word, acronym, link, emoji, or punctuation mark in tweets.

Text cleaning is the second phase and is responsible for eliminating any unnecessary textual material from the tweet content. The third stage is Parts of Speech (PoS) labelling, which extracts the portion of speech tags from the incoming text.

3.4 **Feature** extraction

Following the application of preprocessing, tweets are transformed into the feature matrix by computing the eleven characteristics listed below from the Twitter database [23].

- It shows the total number of words accessible in the tweets.
- Positive Emojis: It, such as :), ;), :D, etc., is employed to convey joyous occasions. This feature counts the total number of positive emojis in tweets using a good emoticon lexicon.
- Negative Emoji: It is the particular symbols used to indicate sad or negative emotions, such as (,: ′(, >: (, etc. A vocabulary of negative emoticons is used to figure out how many negative emojis are used in tweets.
- Neutral Emoji: It (emoji with a neutral expression) does not convey any specific emotion. Total neutrality emojis are calculated by matching tweets to a vocabulary of neutral emoticons.
- Positive Interjection: Words with exclamation marks, such as "Hooray!" and "Wow!" can be used to express a very strong sentiment or attitude toward the issue. Positive exclamation vocabulary is employed to count positive interjections for the same reason.
- Negative Exclamation: Negative expletives are tallied by matching the tweet to a lexicon of negative expletive marks.
- Negation: To indicate a negative viewpoint, negation terms like "no," "not," etc. are often used. This function recognizes the number of negation terms in a tweet by matching it to a list of negation keywords.
- Positive keywords: This function counts the number of positive phrases such as "accomplish," "have confidence," and so on. If there are two negative phrases, these keywords are seen as a single good word. This is called "double positivity."
- Negative terms: This feature displays the overall counts of negative keywords in tweets, such as "terrible," "lost," etc.
- Neutral phrases (okay, seldom) do not evoke any specific emotion or sensation. Matching tweets to a lexicon of neutral terms yields the total number of neutral keywords.
- Intense adjectives: Intense texts, such as "very," "a lot," etc., are employed to make a phrase more powerful or intense. The total number of strong words is computed using a lexicon of intense terms.

Furthermore, the existence of sarcasm or irony in tweets may diminish the value of the previously described traits. The suggested technique employs explicit incoherence, implicit congruency, pragmatic aspects (smiles, emojis, etc.), and exaggeration features (exclamation, quotations, etc.) to address the issue of sarcasm or irony in the Twitter database. The existence of both positive and negative polarisation terms in a tweet indicates explicit incoherence, particularly if the post has an initial positive polarization. For instance, "I like being irritated," where "like" is a favourable word and "irritated" is a negative term. To identify this sort of tweet, positive and negative keywords are tallied in addition to their order, and the featured value of negative keywords rises as a result. Additionally, some tweets have a negative phrase before a positive one, such as "I despise Usain Bolt since he constantly wins." These messages seem unpleasant, yet they are positive. In this situation, the overall count of positive and negative keywords is determined, and if the numbers are identical and a positive keyword precedes a

negative keyword, the score of the positive feature keyword is increased.

The Cuckoo Optimising Approach (COA) is the most common and effective optimization technique [24]. It is influenced by a cuckoo bird's behaviour. They may deposit their eggs in other species nests. It is optimised to handle a variety of difficulties, such as energy dispatching, controller variables, cluster computation, system price, and ease of accessibility, and some limitations are specified. The COA method addresses scheme integrity optimization by utilising heterogeneous elements. Eggs are placed in the nests of other creatures, which may be manufactured as a possible solution. This is elaborated as given in the following steps.

Step 1: Initialize the variables, including the greatest generation of cuckoos (N_g) and the number of nests to be investigated (M).

Step 2: Construct the nest. The nest may be constructed in the following manner using Equation (1).

$$\begin{aligned} N_1(s,m) &= \{s_1, s_2, \cdots, s_n, m_1, m_2, \cdots, m_n\} \\ N_2(s,m) &= \{s_1, s_2, \cdots, s_n, m_1, m_2, \cdots, m_n\} \\ &\vdots \\ N_{T-1}(s,m) &= \{s_1, s_2, \cdots, s_n, m_1, m_2, \cdots, m_n\} \\ N_T(s,m) &= \{s_1, s_2, \cdots, s_n, m_1, m_2, \cdots, m_n\} \end{aligned} \quad (1)$$

where $N(s,m)$ denotes a set of possible solutions. The bird and the eggs are denoted $s_x \; and \; m_x$. The total number of nests is expressed N_T.

Step 3: The penalty feature that implements the restriction is depicted in Equation (2).

$$\hat{T}_s(s,m) = S_s(s,m) + \propto_1 \times \max\{0, f_1(s,m) - V\} + \propto_2 \times \max\{0, f_2(s,m) - C\} + \propto_3 \times \max\{0, f_3(s,m) - W\} \quad (2)$$

The bird and the eggs are denoted $s \; and \; m$, and the scaling factors are denoted $\propto_1 \; and \; \propto_2$. The optimization functions for nest, bird and eggs are expressed $f_1, f_2, and \; f_3$.

Step 4. The cuckoo's egg might be positioned by the novel's COA is expressed in Equation (3).

$$ELR = \beta \times \left\{\frac{N_{c_{egg}}}{N_{egg}}\right\} \times \{V_{hi} - V_{low}\} \quad (3)$$

where ELR indicates the Egg Laying Radius, β indicates an integer number, and V_{hi} and V_{low}, correspondingly, represent the upper and lower bounds. The current number of eggs and the total number of eggs are represented $N_{c_{egg}} \; and \; N_{egg}$.

Step 5: The cuckoo's egg is presented with three distinct hosts and options. Therefore, the cuckoo egg includes three distinct chances for successful development, designated host quality δ_1, δ_2, and δ_3. This value is determined individually by each generation. The nest is now divided into three groups, G_1, G_2, and G_3, and these quantities are personal. The characteristics of a good host are represented in Equation (4).

$$\begin{aligned} &G_1 \; nests \; with \; \delta_1 \quad where \; G_1 \in \{G\} \\ &G_2 \; nests \; with \; \delta_2 \quad where \; G_2 \in \{G - G_1\} \\ &G_3 \; nests \; with \; \delta_3 \quad where \; G_3 \in \{G - G_2 - G_1\} \end{aligned} \quad (4)$$

Step 6: The ideal generation of cuckoos goes to different habitats, and the best answer in the next generation is employed to improve the search solution.

Step 7. Repeat Steps 2–6 until the desired number of generations, N_g, has been attained.

The optimum solution for feature selection using Cuckoo's search algorithm is shown in Algorithm I, as given below.

Algorithm I
Input – F, $\boldsymbol{N_g}$
Initialisation
Start
While $\boldsymbol{z < N_g}$
Generate nests using $\{\boldsymbol{N_1(s,m), N_2(s,m), \cdots, N_T(s,m)}\}^{\boldsymbol{T}}$
Evaluate the fitness function **F(x)**
Execute the egg-laying **(ELR)**
Execute the chick phase
Group the cuckoos
Stop
Output – Obtain the optimum solutions

The current nest is indicated by z, the optimization function is denoted by F, and the nest optimization count is denoted by N_g, and Algorithm I describes the COA procedure. To improve classification accuracy, the COA method calculates a fitness value. It generates a positive integer to characterise the potential solution. The fitness value is the minimization of the categorization failure rate. The optimal solution has a lower failure rate, whereas the worst option has a higher error rate. The optimal function is expressed in Equation (5).

$$f(p_x) = ER(p_x) = \frac{N_{mc}}{N_s} \times 100 \quad (5)$$

The error rate concerning the present egg is denoted $ER(p_x)$, the misclassified samples are denoted N_{mc}, and the total number of samples is denoted N_s.

3.5 Feature selection

Diverse automatic and manual methods have been employed to develop sentiment classification characteristics, but subset selection strategies have

received little attention recently. Several strategies have shown the importance of feature minimization in enhancing classification performance.

This section presents a genetic approach to feature selection to enhance the sentiment categorization of blog posts. A Genetic Algorithm (GA) is an evolutionary programming technique that has been employed in a variety of applications for selecting characteristics. As previously stated, they conducted experiments using the most common machine-learning approaches for the Selection Approach (SA) to select the best classification and feature pairing for unigrams.

Due to their capacity to use accumulated knowledge about an originally unknown search area to bias future searches towards viable subspaces, evolutionary operators have exhibited significant improvements over a range of randomised and localised search techniques. Since evolutionary operators are essentially a domain-independent search approach, they are appropriate for situations where it is challenging or impossible to provide domain theories and knowledge.

A Steady-State Genetic Algorithm (SSGA) operates on a set of chromosomes. The population is subjected to function assessment, crossovers, and mutation procedures until the optimal solution is obtained. In this part, the stages of feature selection using evolutionary algorithms are presented. The steady-state genetic algorithm is shown in Algorithm II.

Algorithm II
Step 1: population initialization (P)
Step 2: Sort participants in P using fitness function
Step 3: do for generation count less than required generation
Choose two parents $\boldsymbol{p_1}$ and $\boldsymbol{p_2}$
Calculate children $\boldsymbol{c_1, c_2} = \boldsymbol{crossover}\ (\boldsymbol{p_1, p_2})$
Compute mutation of $\{\boldsymbol{c_1, c_2}\}$
Accumulate $\boldsymbol{c_1, c_2}$ into population
Do population sorting
Remove participants with lowest fitness function from population P
Repeat step 3 until the required result obtained

3.5.1 Initial population and solution representation

Capturing characteristics is a crucial step in developing an evolutionary algorithm. It uses all the words in the dataset, or unigrams, as characteristics. An answer is expressed by a binary vector whose length is equal to the length of all the words; therefore, the length of all answers is the same. Each solution comprises a subgroup of the beginning word list. For each word belonging to a specific answer, its location in the optimizer vector is linked with the Boolean value 1. The controlled population initialization of the generation algorithm is shown in Algorithm III.

Algorithm III
For every x from 1 to N do
Create a chromosome C
If C is present in the population P
Recreate the chromosome C
End if
Accumulate chromosome C into population P

3.5.2 Evaluation function

In SSGA, the optimization algorithm, or fitness, specifies the quality of an answer. The research evaluated the answers to the challenge based on the precision acquired via machine learning and by examining just the solution-specific terms. The steps for assessing a solution stored in a Boolean matrix are as follows:

- Making an index file with only the keywords from the answer
- Dividing the database into two parts: 75% for modelling learning and 25% for testing.
- Training a strategy using the SSGA method on the trained database with just the keywords considered from the training dataset
- Testing the trained strategy on the testing database using just the solution's keywords
- Optimising the fitness value, which in this case is the model's precision.

The research picked SSGA because of its speed and dependability in producing outcomes. Therefore, the same outcomes are always obtained from identical solutions.

3.5.3 Generating initial population

A random technique is used to produce the starting population. In this stage, each answer is impacted by a subset of the starting collection of keywords. The number of participants is provided to the method as a variable. Each answer is expressed by a Boolean matrix, and the starting community is formed by setting all vector members to zero. Furthermore, for each keyword, a list of all answers that include the keyword is created. This is accomplished by giving the value 1 to the index of the keyword for each answer array in the list.

Assume, the system has a collection of three keywords $[w_1, w_2, w_3]$ and the desire to construct an initial community of four people $[p_1, p_2, p_3, p_4]$. For w_1, a randomised list $[[p_1, p_2, p_4]$ is formed; for w_2, a randomised list $[p_2, p_3, p_4]$ is produced. For w_3, a randomised list is constructed $[p_3, p_4]$. The research obtained $p_1 = [w_1], p_2 = [w_1, w_3], p_3 = [w_2, w_3], p_4 = [w_1, w_2, w_3]$ denoted as $p_1 = \{1,0,1\}, p_2 = \{1,1,1,\}, p_3 = \{0,0,1\}, and\ p_4 = \{0,0,1\}$ with index $(w_x) = x$ for all individuals.

3.5.4 Fitness performance and selecting procedure

To create the fitness value of the SSGA approach, the research examines three elements: power expenditure (C_c), total fall detecting rate (D_r), and the number of sensors being employed (N_s). The relationship is shown in Equation (6).

$$\frac{E}{V} = Cu_n t_n + Cu_1 t_1 + Cu_k t_k + \sum_{x=0}^{N} Cu_{C_s} t_{C_x} \quad (6)$$

E is the energy, and V is the source power, and Cu_i and t_i denoted the current flow and the simulation period, correspondingly, where "i" could be followed by

- microcontroller operating in regular state m
- the microcontroller in power-saving state l
- the gadget in transmitting state t
- the transmission equipment in mode k of reception
- additional elements c

As this analysis is conducted offline, it is believed that the power usage of the Central Processing Unit (CPU) in all states other than m mode is zero (Cu_1=Cu_2=Cu_k=Cu_c=0). Therefore, the proportional power usage of two characteristic patterns is expressed in Equation (7).

$$\frac{Ex_1}{Ex_2} = \frac{Cu_{n_1} t_{n_1} V_1}{Cu_{m_2} t_{m_2} V_2} \quad (7)$$

The current flow, simulation period and supply voltages are denoted $Cu_{n_x}, t_{n_x, and}\ V_x$. Assuming that these two characteristic patterns are performed using identical sensor equipment, the supplied voltage and current consumption are identical ($Cu_{n_1} = Cu_{n_2}$ and $V_1 = V_2$). Consequently, the research infers that power usage (C_c) is related to operating time (t). The operating time was employed to satisfy the energy usage (C_c) criterion. The SSGA system must have minimal false positivity (false alerts) and minimal false negativity (falls that are not identified). As a result, the F-score was employed to evaluate the classifier's identification rates, as it accounts for both false positivity and false negativity. The F-score is calculated by Equation (8).

$$F = \frac{2T^P}{2T^P + F^N + F^P} \quad (8)$$

T^P stands for true positives, F^P stands for false positives (false alarms), and F^N stands for false negatives (undetected falls). Therefore, the system used the F-score as the total identification rate (D_r) factor. One of the hallmarks of a workable system for recognising human behaviour (C_c) is that users do not need to wear many sensing devices. As a result, the number of sensors used, N_s, was incorporated into the fitness function $(F(i))$. Consequently, the fitness value of this research is computed using Equation (9).

$$F(i) = w_1 D_r - w_2 C_c - w_3 N_s \quad (9)$$

i = a chromosome presents in the populations
D_r = the F-score-measured total identification rate.
w_1 = weighted for precision
C_c= total chromosomal system computing cost
w_2 denotes the overall computing cost factor.
N_s = number of detectors chosen
w_3 = the weight assigned to the number of detectors used

The entire computing cost is analyzed and measured using Equation (10).

$$C_c = w_1 t_1 + w_2 t_2 + \cdots + w_n t_n \quad (10)$$

w_x and t_x are the weights and simulation periods of x. D_r is measured in percentage (%) and C_c is measured in milliseconds (ms). As N_s represents the number of detectors in use, it lacks a unit. Consequently, for deployment, the fitness value is modified and represented in Equation (11).

$$F(i) = \left(\frac{w_1 D_r}{100}\right) - \left(\frac{w_2 C_c}{2.4}\right) - \left(\frac{w_3 N_s}{3}\right) \quad (11)$$

2.4 and 3 represent the quickest time (in milliseconds) for the machine (a Personal Computer (PC)) to retrieve all characteristics from all detectors and the maximum number of sensors used in this investigation, respectively. w_1 is set to 1, although w_2 and w_3 are both set to 0.5. This is because precision time (D_r) is more significant than power (C_c) and detector count (N_s). Assuming that the relevance of detector locations is the same, an identical value is assigned to 1. Consequently, the optimal sequence is one with the best identification rate, the smallest computing price, and the fewest sensors utilised. These weights may be modified to meet the requirements of the activity.

A roulette-wheel strategy is applied to pick parents from the community, is selected so that superior chromosomes or individuals have a greater chance

of getting selected. The likelihood P(x) of picking the xth object from a group of n elements is weighed more heavily for objects with lower numbers based on Equation (12).

$$P(x) = \frac{k(1-k)^{x-1}}{1-(1-k)^n} \qquad (12)$$

where k = 1/4 is the likelihood of choosing the first item for an unlimited number of items (x). $P(x+1) = (1-k)P(x)$ ensures that the total of this equation equals $\sum_{x=1}^{N} P(x) = 1$ while reducing the chance proportionally for succeeding items (x). It is feasible to determine the reverse of the likelihood of mass dispersion to create random values based on samples from these distributions. Putting $D_r = 1-(1-k)^n$, the likelihood function is expressed in Equation (13) and the inverse function is expressed in Equation (14).

$$Q(l) = \sum_{x=0}^{l} k(1-k)^{x-1} | D_r = \frac{1}{D_r(1-(1-k)^l)} \quad (13)$$

$$l(Q) = \frac{\log(1-D_r Q)}{\log(1-k)} \qquad (14)$$

The simulation duration is denoted D_r, the likelihood function is denoted Q, and the probability is denoted k.

3.5.5 Crossover

Crossover permits the modelling and replication of solutions to generate new ones. Randomly selecting individuals for crossover prevents the system from prematurely converging. Each algorithmic iteration chooses several people for crossover. This value is determined by a variable representing the crossover frequency. Creating a list of eight to fifteen crossover factors (between 0 and the chromosome sizes) is the first step in crossing two people. The child is produced by replicating the divisions of both parents among their respective crossing places. The mutation process is expressed in Algorithm IV.

Algorithm IV

For every gene g in the chromosome C do
Create a randomized number r from 0 to 1
Check gene g=0 and number r < population P
Gene g =1
Check gene g=1 and number r< population p
Gene g=0
End
End for

In Fig. 2, the case on the left uses a singular crossing point (position 3), but the case on the right uses two points (positions 1, and 4). In the SSGA approach, a constraint has been added to the crossover procedure: at least one of the children must be as excellent as their parents; alternatively, the procedure is continued until the requirement is met or a limited number of iterations is achieved. This enables quick improvements in outcomes.

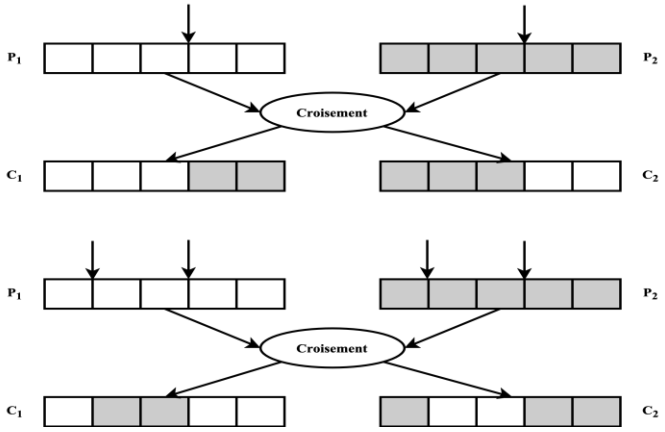

Fig. 2. Cross-over and mutation function

3.5.6 Mutation

The mutation is employed to prevent solutions from converging on a locally optimal state. As with crossover, mutation involves the random selection of participants. The mutation rate permits determining the likelihood of iteratively modifying solutions. A chosen person is mutated by randomly inverting the results of certain chromosomes. The number of modifying genes is often low and is provided as an algorithm variable. This element of an evolutionary algorithm permits the exploration of novel answers and population diversification.

3.5.6 Reinsertion of new population

In this stage, offspring resulting from repeated crossing and mutation are returned to the population, displacing a portion of their parents' offspring in the following generation. Some of the best participants (parents and children) are kept in the community of the next generation, while other people are chosen at random.

The research has picked convergence of the fitness value as the termination criterion. If the optimal answer has not changed after a parameterized number of epochs, the method terminates and returns the optimal solution. This component of an evolutionary algorithm permits the exploration of alternative answers and community diversification. SSGA has several features that outperform GA, including:

- The capacity to discover remedies by using tiny amounts is contradictory to GA, whereas SSGA can choose a good new participants as soon as they are produced, so GA must investigate all inhabitants to select excellent players.
- The capacity to find remedies by using small specimens is contrary to GA. Because SSGA prevents the community from repeatedly including the same person, each generation will have a diverse assortment of solutions.
- SSGA has a greater resistance to genetic differentiation than other varieties. For instance, if we assume that a person is built from healthy genes, we may deduce that this human has a high fitness rating. The crossover operators will be used to create new humans, whose DNA will be completely rearranged. In contrast to GA, the original person, also known as the parents, will continue to exist as a member of the present generation and will retain the ability to be chosen for another crossing.

3.6 Classification model for sentiment analysis

In the training stage, the primary step of the suggested approach is to develop a classifier model that can effectively distinguish between positively and negatively tagged tweets.

Many machine-learning methods might be used to construct such a system. The researchers constructed a classification utilising ANN, BERT [18], TF-IDF [16] and FFBPNN as the basis for learners to compare with the suggested model. These methods are widely used and have a considerable amount of success with text categorization issues.

The suggested model uses groups of simple majority classifiers, which are shown in Fig. 3.

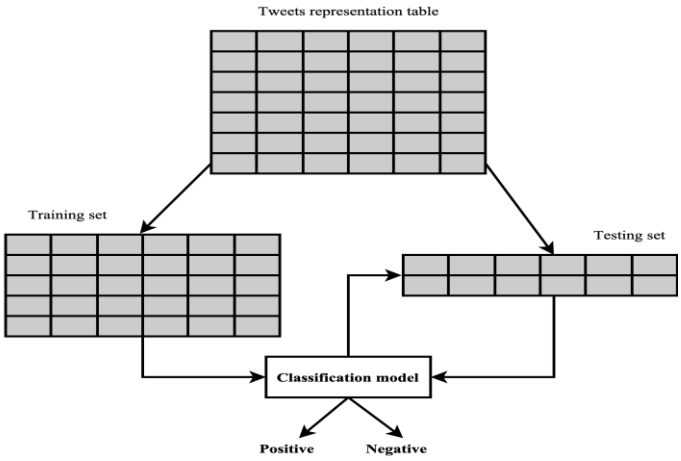

Fig. 3. The Twitter sentiment classification model

As seen in Fig. 3, the extracted characteristics from the input sample of tweets are split into two groups: training and testing. Each classification is given the training dataset to record its judgement. The output

of the voting ensembles is thus the simple majority derived from the three classifications. This tweet will be considered favourable by the optimization method since this is an overwhelming opinion. Eventually, the testing set is utilised to assess the

classification model's efficiency. The proposed HOM-TPA system is compared with the existing models like ANN, BERT [18], TF-IDF [16] and FFBPNN [28].

3.6.1 Naïve Bayes

Let $X = x_1, x_2, \cdots, x_m$, represent the test sample and let $C_1, C_2, \cdots, C_N$ represent the specific class. The sample for classification which is selected is shown in Equation (15).

$$P(X|C_k) = \frac{P(X|c_k)P(P(C_k))}{P(X)} \quad (15)$$

The probability is computed using Equation (3) and the function is expressed as $P(X|C_k)$. For each category, the dataset with the greatest computed likelihood function belongs to that category. The conditional probability of the Cuckoo with the sample is denoted $P(X|c_k)$, and the probability of

the Cuckoo is denoted $P\big(P(C_k)\big)$, and the present sample is expressed $P(X)$.

3.6.2 Support Vector Machine

The SVM is a vector space-based data collection technique that finds the decision boundaries between two categories based on training examples that are farthest from a randomised point. In statistical learning theory, the SVM's hierarchical risk reduction is an intriguing characteristic. SVM is employed for categorization in the suggested study.

SVM was started for the first time to handle binary categorization issues, and SVM's primary purpose is to construct a hyperplane that optimises the separation between two categories. SVM can handle both regular, simple, and complicated categorization jobs. In addition, it can manage solvable and nonsolvable problems in linear and exponential situations. The primary objective is to translate the initial data values from an input field to a high-dimensional input matrix like X. Utilizing kernel functions depending on the empirical learning approach and minimising nonlinear programming difficulties, the translation is done. It has heuristic methods in addition to the various basis functions.

Theoretically, the SVM consists of building a function that correlates to an outcome. The output and the functional relationships between SVM components are denoted in Equations (16a), (16b), and (16c).

$$y = \{-1,1\} for\ each\ input\ X \in R^D \quad (16a)$$

$$f: R^D = \{-1,1\} \quad (16b)$$

$$X \rightarrow y \quad (16c)$$

The input sample is denoted X, and the output is expressed y. The dataset is expressed with D-dimensional input as R^D. As a result, by linearly combining an input signal, a discriminating function h is created. As vector $= x_1, x_2, \cdots, x_m$. The hyperplane solution is expressed in Equation (17).

$$h(X) = wX + k \quad (17)$$

X denotes the input, w represents a vector orthogonal to the hyperplane, k represents a bias variable and $h(X)$ represents the hyperplane solution. The category of X is decided by the signature of $h(X)$, if $h(X) > 0$, then X is a subclass of Class-1; alternatively, X is a subclass of Class-2.

3.6.3 Feed forward back propagation neural network

FFBPNN was utilised for classification. It consisted of one input node, two hidden intermediary levels, and one output unit. The number of synapses in the initial and secondary intermediate levels was determined by trial and error to be 50 and 30, respectively. The output level has eight synapses, which indicate seven kinds of fruit abnormalities in addition to the calyx. Fig. 4 illustrates an instance of a feed network with two hidden levels.

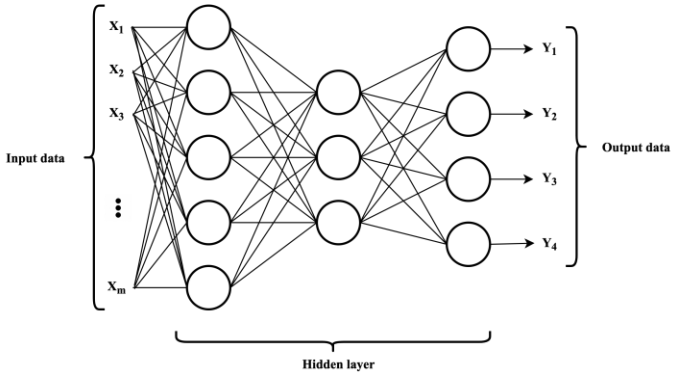

Fig. 4. Architecture of the FFBPNN model

In the network seen in Fig. 4, each synapse in a layer gets data from the synapses of the preceding layer, processes it, and then transmits it to the synapses of the following layer. In this system, X symbolises the input data matrix, Y symbolises the output information matrix, and F is the function that analyses the information. When provided with a sample of positive and negative categories, this system is intended to detect the problem category of orange.Thus, the proposed HOM-TPA model is intended to analyse Twitter polarity. The GA is used to pick features, while the COA approach is used to extract features. The suggested HOM-TPA is examined, and the results are compared to known classification algorithms such as Artificial Neural Networks (ANN), BERT [18], TF-IDF [16] and FFBPNN. The experimental results are evaluated in the next section.

4. Experimental outcomes and findings

The suggested system is constructed, and its performance is evaluated using many well-known databases in the field of Twitter sentiment classification. The methods of preprocessing and extraction of features are accomplished in Java using the Stanford CoreNLP package. The RapidMiner® tool is used to select and categorise features using the suggested HOM-TPA method.

4.1 Datasets

To analyse the effectiveness of the developed framework, four databases are used to assess the intended operations. The Stanford Twitter Sentiment Corpus comprises around 1.6 million tweets (800,000 positive and 800,000 negative tweets) gathered using a scraper that queries the Twitter Application Programming Interface (API) [29]. Due to technological restrictions, the research did not utilise the whole training database in the experimental analysis. The research uses unified selection to generate two sample databases, Stanford-1K and Stanford-3K, containing 1000 and 3000 tweets, respectively.

Sanders Dataset contains approximately 5000 tweets that were manually labelled as "good," "negative," "neutral," and "unimportant." [30] Four search keywords are utilised on the Twitter website: @apple, #google, #microsoft, and #twitter. They were unable to collect all these tweets since the majority are presently invalid or removed. They are only concerned with tweets that have been tagged as positive or negative, which amounts to around 201 positive and 293 negative posts. The Health Care Reform (HCR) Database is the title of the fourth database. This database was produced by crawling March 2010 tweets using the hashtag "#hcr". Several of these posts have been classified as "good," "bad," or "neutral." The research is only concerned with positive and negative tweets, which total around 630 (250 positive and 380 negatives). The true positive and true negative is expressed T^P *and* T^N, and false positive and false negative are expressed F^P *and* F^N.

Table 1. Description of the databases

Dataset	No. of tweets	
	Positive	Negative
Standford-1K	450	450
Standford-3K	1000	1000
Sanders	200	200
Health Care Reform	250	380

The database and its details for all four databases are expressed in Table 1. The number of favourable and negative tweets analysed is shown in Table 1. The Standford-1K dataset has 450 positive and 450 negative tweets; the Standford-3K dataset contains 1000 positive and 1000 negative tweets; the Sanders dataset contains 200 positive and 200 negative tweets; and the Health Care Reform dataset contains 250 positive and 380 negative tweets.

4.2 Experimental analysis

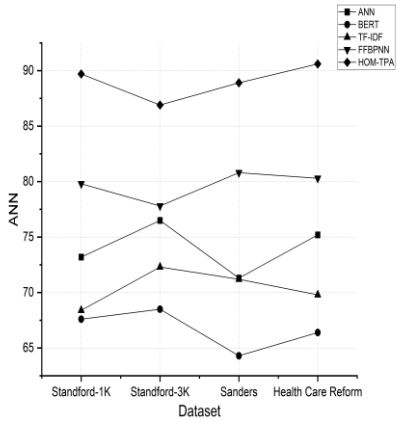

Fig. 5. The accuracy evaluation of the HOM-TPA model

Analyses are performed on the accuracy assessment of the proposed HOM-TPA model, and the findings are compared with the results obtained by other classification models, such as ANN, BERT [18], TF-IDF [16] and FFBPNN[28]. The accuracy is computed using Equation (18).

$$A = \frac{T^N + T^P}{Total\ samples} \quad (18)$$

During the experimental research, four distinct datasets were taken into consideration. The relative

precision of the various approaches is dissected and evaluated before being compared to one another. The findings validate the usefulness of the suggested HOM-TPA model, which uses the SSGA for feature selection and the cuckoo search method for feature extraction, respectively. The overall accuracy of the proposed HOM-TPA model is improved by 24.2% because of the combined outcomes of feature selection and extraction when compared to the accuracy of the current models.

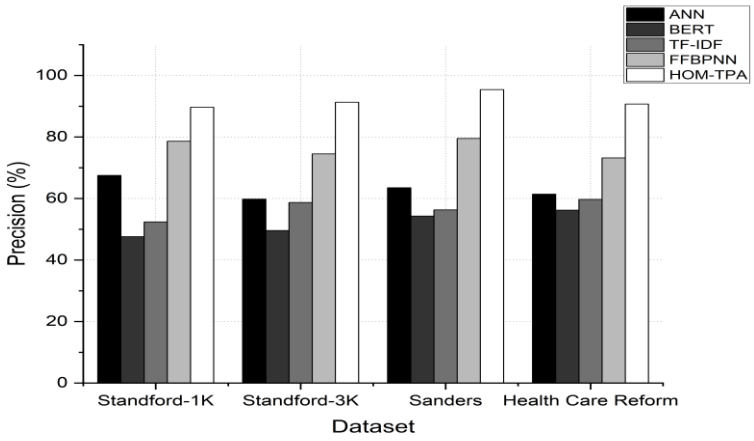

Fig. 6. Precision evaluation of the HOM-TPA model

The precision assessment of the proposed HOM-TPA model is analysed across a variety of datasets, and the findings are compared to other classifiers that are already in use. Figure 6 illustrates the findings of the study. The precision is computed using Equation (19).

$$P = \frac{T^P}{T^P + F^P} \quad (19)$$

The HOM-TPA model that was proposed is used in the analysis of Twitter polarisation from users, and the system demonstrates higher precision in detecting and classifying Twitter polarisation by employing the SSGA for feature selection and the cuckoo search algorithm for feature extraction. Both algorithms are utilised by the system. That increases the results by 28.5% than the existing system. The combined findings assure superior performance compared to the classifiers that were previously used.

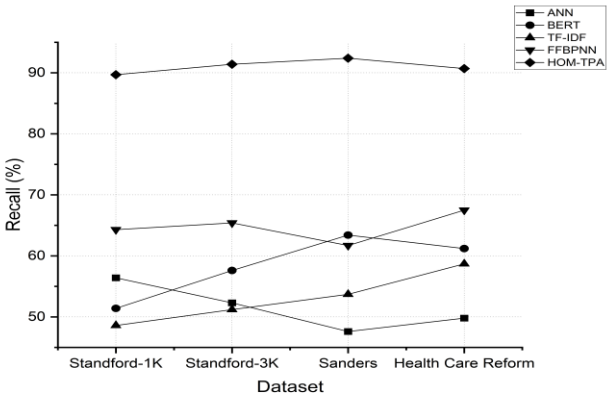

Fig. 7. Recall analysis of the proposed HOM-TPA model

The recall performance of the new HOM-TPA model is compared with the current ANN, BERT [18], TF-IDF [16] and FFBPNN classifiers and four distinct datasets are used to analyse the suggested model's performance. The comparisons of the findings are also illustrated graphically in Figure 7. The recall function is computed using Equation (20).

$$R = \frac{T^P}{T^P + F^N} \quad (20)$$

The HOM-TPA model that was proposed beats all the other classification models that are currently available thanks to hybrid optimization approaches that use feature selection and feature extraction methods. When compared to the performance of a

single machine learning method, the combined results for Twitter polarity classification are superior. The results obtained has 45.3% more than the existing models.

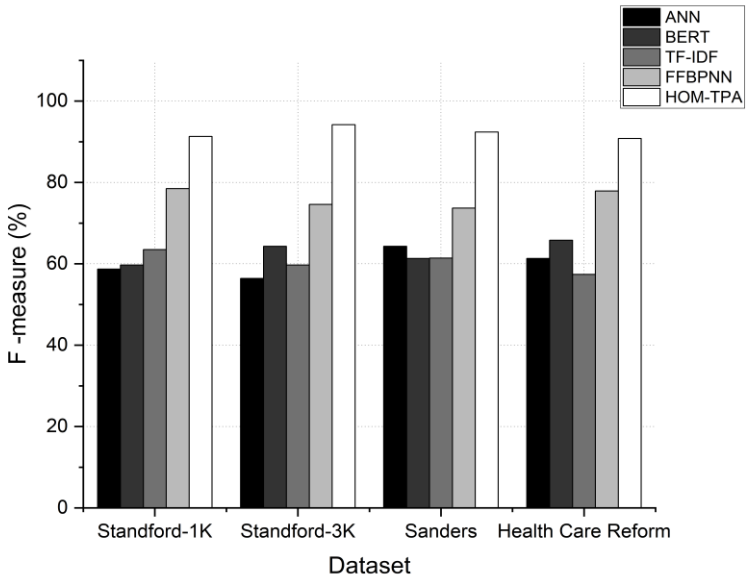

Fig. 8. F-measure comparison of the proposed HOM-TPA model

In this study, both the suggested HOM-TPA model and the experimental data in terms of F-measure are dissected, evaluated, and shown in Fig. 8. Using a variety of Twitter datasets, the analysed results of the proposed HOM-TPA model are compared with the categorization methods that are already available. The F-measure is analysed and shown in Equation (21).

$$F = \frac{2*P*R}{P+R} \quad (21)$$

The precision and recall are expressed P and R, and the values of the functions are computed using Equations (19) and (20). The suggested HOM-TPA model, combined with the SSGA algorithm and the cuckoo search algorithm, has produced cumulative results that demonstrate improved feature selection and feature extraction models. The suggested HOM-TPA model achieves results with a percentage greater than 90 per cent across all four distinct Twitter datasets.

Table 2. Mean Absolute Error performance comparisons

Dataset	NB	SVM	ANN	BERT	TF-IDF	FFBPNN	HOM-TPA
Standford-1K	28.5	42.6	24.3	19.8	21.5	14.3	12.3
Standford-3K	27.5	37.4	21.3	18.6	19.5	17.5	10.6
Sanders	32.4	32.4	19.6	19.4	19.2	18.5	11.5
Health Care Reform	26.9	29.6	24.3	18.2	18.6	16.3	9.4

The proposed HOM-TPA model's Mean Absolute Error (MAE) is analysed under four distinct datasets, and the comparison results with other existing classifiers are provided in Table 1. The mean absolute error is computed using Equation (22).

$$MAE = \frac{\sum_{i=0}^{N-1} y_i - x_i}{N} \quad (22)$$

The output and the input are denoted x_i *and* y_i, and the total number of samples is expressed N. The HOM-TPA model that was suggested has a GA for feature selection and a COA for feature extraction, which results in a decreased MAE. When compared to the NB, SVM, ANN, BERT [18], TF-IDF [16] and FFBPNN [28] classifiers that are already in use, the mean absolute error produced by the hybrid optimization approach is much smaller. An analysis is done on the tweets, and the HOM-TPA model demonstrates more accuracy in polarity analysis which is 42.6% than the existing models.

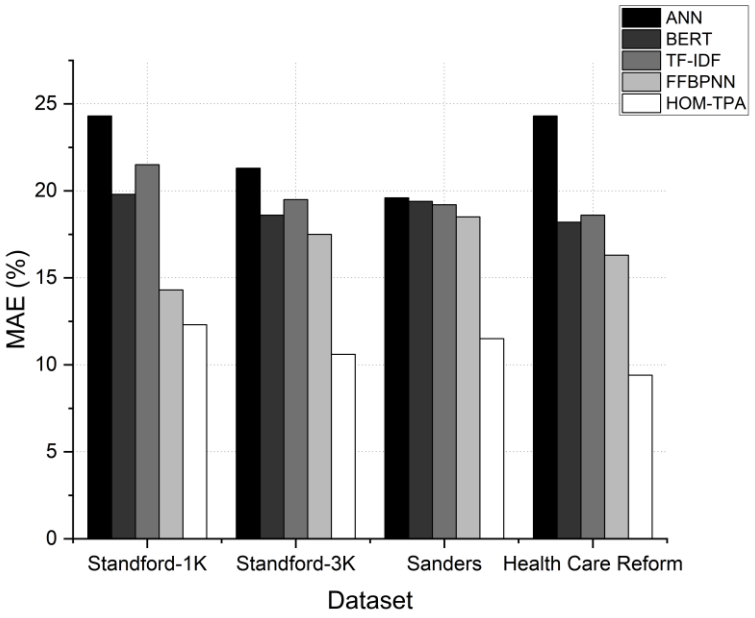

Fig. 8. Mean Absolute Error comparison of the proposed HOM-TPA model

The MAE of the proposed HOM-TPA model is analysed using four separate datasets, and the comparative results with other current classifiers are shown in Fig. 8. Because using a GA for feature selection and a COA for feature extraction in the HOM-TPA model that was recommended, the MAE was found to be significantly reduced. The MAE that is generated by the hybrid optimization strategy is much lower in comparison to the mean absolute error that is generated by the ANN, BERT [18], TF-IDF [16], and FFBPNN classifiers that are currently being used. The tweets are subjected to analysis, and the results show that the HOM-TPA model provides a higher level of accuracy (42.6%) in polarity analysis.

5. Conclusion and the findings

Presently, Twitter sentiment analysis is one of the emerging study fields for identifying and analysing the perspectives and emotions of people. In this academic research, a novel approach to the categorization of tweet sentiment categories is established. A Hybrid Optimization Model for Twitter Polarity Analysis (HOM-TPA) is designed in this article. The work employs a steady-state genetic algorithm for feature selection and a cuckoo search for feature extraction. This experimental investigation was validated against a publicly accessible database, twitter-sanders-apple2, demonstrating the utility of the suggested technique. Even though the suggested approach is more accurate than the current methods, more accuracy enhancement is obtained using the hybrid optimization method. The results are verified using experimental analysis using four different datasets, and the results are measured in terms of accuracy, precision, recall, F-measure, and MAE. Furthermore, future studies will investigate the possibility of improving accuracy by incorporating a feature selection approach and using other optimization technique variations. In addition, there is room for improvement in dealing with sarcastic and ironic tweets. In addition, domain-specific ontologies and contextual data at the keyword and post levels may be utilised for classifying tweets. Future research may involve including "neutral" tweets in the suggested approach by modifying the feature extraction and classification procedures to effectively identify these tweets. A multi-objective categorization model is paired with an efficient feature selection strategy to improve the precision of Twitter sentiment assessment.

References

1. Kumar, A., & Garg, G. (2019). Sentiment analysis of multimodal Twitter data. Multimedia Tools and Applications, 78, 24103-24119.
2. Balaji, T. K., Annavarapu, C. S. R., & Bablani, A. (2021). Machine learning algorithms for social media analysis: A survey. Computer Science Review, 40, 100395.
3. Yadav, A., & Vishwakarma, D. K. (2020). Sentiment analysis using deep learning architectures: a review. Artificial Intelligence Review, 53(6), 4335-4385.

4. Zhou, J., Huang, J. X., Chen, Q., Hu, Q. V., Wang, T., & He, L. (2019). Deep learning for aspect-level sentiment classification: survey, vision, and challenges. IEEE Access, 7, 78454-78483.
5. Jain, A., & Jain, V. (2019). Sentiment classification of Twitter data belonging to renewable energy using machine learning. Journal of information and optimization sciences, 40(2), 521-533.
6. Lopez, C. E., & Gallemore, C. (2021). An augmented multilingual Twitter dataset for studying the COVID-19 infodemic. Social Network Analysis and Mining, 11(1), 102.
7. Pota, M., Ventura, M., Catelli, R., & Esposito, M. (2020). An effective BERT-based pipeline for Twitter sentiment analysis: A case study in Italian. Sensors, 21(1), 133.
8. Nezhad, Z. B., & Deihimi, M. A. (2022). Twitter sentiment analysis from Iran about COVID-19 vaccine. Diabetes & Metabolic Syndrome: Clinical Research & Reviews, 16(1), 102367.
9. Rehioui, H., & Idrissi, A. (2019). New clustering algorithms for Twitter sentiment analysis. IEEE Systems Journal, 14(1), 530-537.
10. Bibi, M., Aziz, W., Almaraashi, M., Khan, I. H., Nadeem, M. S. A., & Habib, N. (2020). A cooperative binary-clustering framework based on majority voting for Twitter sentiment analysis. IEEE Access, 8, 68580-68592.
11. Wang, L., Niu, J., & Yu, S. (2019). SentiDiff: combining textual information and sentiment diffusion patterns for Twitter sentiment analysis. IEEE Transactions on Knowledge and Data Engineering, 32(10), 2026-2039.
12. Naseem, U., Razzak, I., Musial, K., & Imran, M. (2020). Transformer-based deep intelligent contextual embedding for Twitter sentiment analysis. Future Generation Computer Systems, 113, 58-69.
13. Pota, M., Ventura, M., Catelli, R., & Esposito, M. (2020). An effective BERT-based pipeline for Twitter sentiment analysis: A case study in Italian. Sensors, 21(1), 133.
14. Saad, S. E., & Yang, J. (2019). Twitter sentiment analysis based on ordinal regression. IEEE Access, 7, 163677-163685.
15. Alharbi, A. S. M., & de Doncker, E. (2019). Twitter sentiment analysis with a deep neural network: An enhanced approach using user behavioural information. Cognitive Systems Research, 54, 50-61.
16. Neogi, A. S., Garg, K. A., Mishra, R. K., & Dwivedi, Y. K. (2021). Sentiment analysis and classification of Indian farmers' protest using Twitter data. International Journal of Information Management Data Insights, 1(2), 100019.
17. Zhang, Y., Song, D., Zhang, P., Li, X., & Wang, P. (2019). A quantum-inspired sentiment representation model for Twitter sentiment analysis. Applied Intelligence, 49, 3093-3108.
18. Azzouza, N., Akli-Astouati, K., & Ibrahim, R. (2020). Twitterbert: Framework for Twitter sentiment analysis based on pre-trained language model representations. In Emerging Trends in Intelligent Computing and Informatics: Data Science, Intelligent Information Systems and Smart Computing 4 (pp. 428-437). Springer International Publishing.
19. Kumar, A., & Garg, G. (2019). Sentiment analysis of multimodal Twitter data. Multimedia Tools and Applications, 78, 24103-24119.
20. Sailunaz, K., & Alhajj, R. (2019). Emotion and sentiment analysis from the Twitter text. Journal of Computational Science, 36, 101003.
21. Parmar, M., Torper, O., & Drouin-Ouellet, J. (2019). Cell-based therapy for Parkinson's disease: A journey through decades toward the light side of the Force. European Journal of Neuroscience, 49(4), 463-471.
22. Ramachandran, D., & Parvathi, R. (2019). Analysis of Twitter-specific preprocessing techniques for tweets. Procedia Computer Science, 165, 245-251.
23. Kumar, A., & Garg, G. (2019). Sentiment analysis of multimodal Twitter data. Multimedia Tools and Applications, 78, 24103-24119.
24. Ragwan, E., Huo, R., & Kung, Y. (2022). Altering NAD (P) H cofactor specificity by structure-guided modification of class II HMG-CoA reductase. The FASEB Journal, 36.
25. Cresci, S., Petrocchi, M., Spognardi, A., & Tognazzi, S. (2019). On the capability of evolved spambots to evade detection via genetic engineering. Online Social Networks and Media, 9, 1-16.
26. Villavicencio, C., Macrohon, J. J., Inbaraj, X. A., Jeng, J. H., & Hsieh, J. G. (2021). Twitter sentiment analysis towards covid-19 vaccines in the Philippines using naïve bayes. Information, 12(5), 204.
27. Hidayat, T. H. J., Ruldeviyani, Y., Aditama, A. R., Madya, G. R., Nugraha, A. W., & Adisaputra, M. W. (2022). Sentiment analysis of Twitter data related to Rinca Island development using Doc2Vec and SVM and

logistic regression as a classifier. Procedia Computer Science, 197, 660-667.
28. Kaur, G., & Kukana, P. (2020, September). Sentiment Analysis using Cuckoo Search and Computational Intelligence. In 2020 International Conference on Smart Electronics and Communication (ICOSEC) (pp. 497-503). IEEE.
29. https://www.kaggle.com/datasets/kazanova/sentiment140
30. https://catalog.data.gov/dataset/?tags=sanders
31. https://archive.ics.uci.edu/ml/datasets/Health+News+in+Twitter