International Journal of Engineering Trends and Technology
ISSN: 2231 – 5381 / https://doi.org/10.14445/22315381/IJETT-V70I10P206

Volume 70 Issue 10, 44-53, October 2022

Original Article

Cuckoo Search Optimized Improved Opinion Mining and Classification

Priyanka[1], Kirti Walia[2]

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

[1]Corresponding Author : priyankatuli1986@gmail.com

Received: 21 July 2022 Revised: 22 September 2022 Accepted: 05 October 2022 Published: 19 October 2022

Abstract *- Opinion mining presents one of the most prominent fields in sentiment analysis to deal with the enormous content generated by social media. Opinion mining is used to track people's moods based on any product and helps collect different reviews against the product in many fields. This paper aimed to identify the individual's emotions and sentiments expressed by their with various subjects and products. The present work comprises pre-processing of the extracted tweets data, followed by TF-IDF-based feature extraction and feature selection, and optimization using the Cuckoo Search algorithm. The sentiment classification is performed using Naïve Bayes and Support Vector Machine into positive, negative, and neutral opinions. In the simulation analysis, 10000 samples were analyzed in terms of precision, recall, and accuracy. The overall analysis shows that CS-SVM outperformed the CS-Naïve Bayes classifier with an average accuracy of 93%. The success of the proposed CS-optimized sentiment classification is further justified by comparative analysis against the existing studies.*

Keywords *- Cuckoo Search, Naïve Bayes, Opinion Mining, Sentiment Analysis, Support Vector Machine.*

1. Introduction

The computational study that is applied to analyze different opinions of people, like appraisals, attitudes, evaluations, appraisals, emotions, etc., against any products, issues, organizations, events, and their attributes are known as opinion mining or sentiment analysis. Opinions are considered a central part of human activities that shows the behavior of making decisions based on others' opinions [1]. In opinion mining, text elements are processed with the help of Natural Language Processing (NLP) which applies math to determine positive or negative. Many techniques are available to analyze the author's view that depends on NLP textual information.

The opinion mining software automatically extracts people's emotions, sentiments, and opinions and also shares people's feelings on the web [2, 3]. The author introduced a Twitter sentiment analysis framework [4]. In recent years, people have been very comfortable sharing their feelings regarding products using social media like blog posts, reviews, comments, and tweets depending on different topics [5]. Opinion mining helps in many fields to collect different reviews against the product. For example, in marketing, if any new product is launched, then opinion mining describes the likes or dislikes of people against the product features [6]. Sentiment analysis determines a piece of writing that belongs positive, negative, or neutral. The sentiments are classified into two positive or negative categories and are also categorized on an n-point scale, i.e., good, satisfactory, very good, bad, or very bad [7].

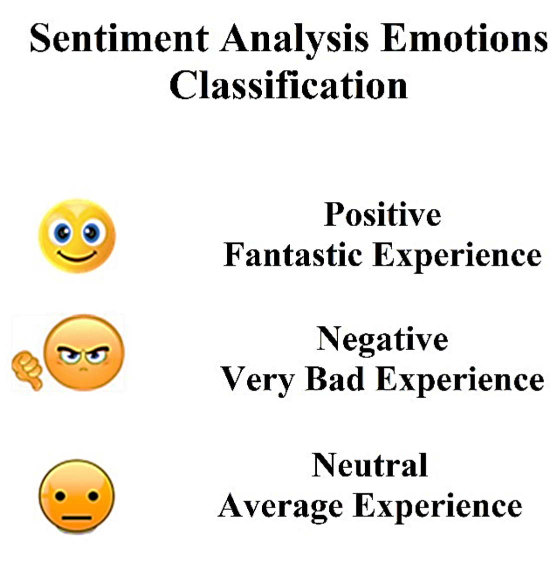

Fig. 1 Sentiment analysis emotions classification

Fig. 1 describes the different sentiment analysis emotions classification that includes three sub-categories positive, neutral, and negative. The positive indicates that users are fully satisfied with the product. The neutral indicates that the product is average means not very good or not very bad. The negative defines that the product is very bad or useless. In recent years, many researchers represented a lot of work based on opinion analysis using the CNN and hybrid techniques using the SVM [8, 9]. Different authors proposed several applications of sentiment analysis. SA is applied to various sectors, such as political, medical, business, public opinion, etc., to collect several opinions of the users [10]. One of the recent research works presented by [11] is based on the sentiment analysis of COVID-19 with the help of Twitter datasets.

The major difficulty arises when encountering sarcastic sentences. They reflect different senses and meanings depending upon the situation of the sender. Also, it has been observed that people express their negative sentiments while using positive words, which significantly challenges the sentiment analysis work. It also adds to the variability among the textual contents. However, most researchers have not included the normalization that is a must when using non-standard data for sentiment classification or opinion mining. Another gap that exists is the difficulty in assigning the polarity to aspects that have observed great bias.

2. Literature Survey

This section of the paper discussed literature surveys that several authors represent by implementing different approaches for sentiment analysis or opinion mining. The proposed work implements Cuckoo Search (CS) for optimization, and Naïve Bayes (NB) is applied for classification.

Kowcika *et al.* (2013) presented a system that collects valuable information from Twitter and analyzed sentiment analysis based on tweets for smartphones conflict. The proposed system effectively counts the age of the user. The Naïve Bayes approach is applied to categorize the category of the user with the help of gender. The tweet has been labeled through a sentiment classifier model that systematically examines the data based on different user constraints such as age, gender, and location group [12].

Pandey *et al.* (2017) introduced a metaheuristic approach named Cuckoo Search, and K-means (CSK) was applied to find out the cluster heads based on sentimental contents. The author performs experimental results on different Twitter datasets. The proposed CSK approach has been compared with other techniques such as Cuckoo Search (CS), Particle Swarm Optimization (PSO), Improved CS, and two N-Grams approaches. The experimental results describe that the CSK performs better than other approaches, and the proposed work accuracy is computed as 79% [13].

Zvarevashe and Olugbara (2018) designed a sentiment analysis with an opinion-mining framework based on the feedback of hotel customers. The author selected the dataset named as OpinRank because it has unlabeled reviews that provide flexibility based on custom experimentation. The proposed work automatically prepared a sentiment dataset for training and testing that helps to extract opinions of the hotel services through reviews. As in the result section, a comparative analysis of NB multinomial, minimal sequential optimization, complement Naïve Bayes (NB), and Composite hypercubes are performed to discover a suitable algorithm for classification. The highest precision is calculated as 80.9% by applying the Naïve Bayes multinomial [14].

Kermani *et al.* (2020) proposed an approach based on machine learning for Twitter sentiment analysis. The author applied method four feature extractions: TF-IDF, scoring using SentiWordNet, sentiment scoring, and semantic similarity that define the tweets class. For classification, two classifiers were applied: SVM and NB. The experimental results show that the proposed work provides higher accuracy than existing approaches [15].

Aljameel *et al.* proposed a model that analyzes sentiment analysis based on the awareness of the preventive procedures for COVID-19. The author implemented the study in the five regions of Saudi Arabia. The Twitter dataset is collected at the time of curfew. Several machine learning algorithms are applied to process the data: SVM, NB, and KNN. The experimental results show that the SVM classifier performs better with the TF-IDF approach and computed higher accuracy of 85%. At last, the author describes that the people who belong to the southern region have the highest awareness of COVID-19, and the people of the middle region have the least awareness [16].

3. Proposed Methodology

This section describes the methodology related to the proposed work. The dataset used for the analysis is "Twitter Sentiment Analysis," available at the Kaggle website https://www.kaggle.com/c/twitter-sentiment-analysis2. It is used for the text processing task to determine positive, negative, and neutral opinions. The steps used in the proposed work are further divided into pre-processing, feature extraction, feature selection, training, and classification. Fig. 2 describes the workflow diagram related to the proposed work.

3.1. Pre-processing

The first step of the methodology section is known as pre-processing. This is the basic step that cleans the collected data. The data is collected from different sources that are in an unorganized form [17]. It contains unwanted information. So, the pre-processing step helps clean the data and organize it. Several authors applied different pre-processing approaches to remove unwanted textual information for sentiment analysis [18, 19, 20]. Krouska *et al*. (2016) introduced several pre-processing techniques applied to the three Twitter datasets related to sentiment analysis. In the proposed work, the major contribution of pre-processing step is categorized into three sub-parts: normalization, punctuation removal, and stop words [30].

3.1.1. Normalization

The first step that comes under the pre-processing is known as normalization, which is applied to remove the irrelevant data from a large collection of the extracted data. When the data is extracted from the datasets, it is full of noise, including URLs, links, tags, etc. So normalization helps remove the data that is unimportant or related to the work. Javed & Kamal applied normalization for sentiment analysis to remove the tags and URLs because these are not important for the sentiment analysis process [22].

3.1.2. Punctual Removal

The text contains several punctuation symbols like commas and colons that do not contain any relevant information. These symbols are not useful for text analysis. So punctuation symbols are removed from the input text [23]. Text-based sentiment analysis also consists of several punctuation marks, so removing all the punctuation marks from the text is very important.

3.1.3. Stop Word Removal

During the pre-processing step, filtered words are known as stop words. Some common words are articles, pronouns, a, and, the, etc. Stop words help to remove unwanted words that are useless for text classification. Stop word removal is also very helpful for sentiment analysis. After all, it helps to enhance the system's performance because it removes all the meaningless words. The illustration of pre-processing is given in Table 1. It describes the pre-processing through the example. The text is processed according to the pre-processing methods. The result of the pre-processing is passed to the feature extraction in which relevant features are extracted.

Table 1. Illustration of Text Pre-processing

Processing	**Example**
Raw Text	Sentiment analysis is a significant task in Natural Language Processing (NLP).
Normalization	Sentiment analysis is a significant task in natural language processing (NLP).
Punctuation Removal	sentiment analysis is a significant task in NLP
Stop Word Removal	sentiment analysis significant task in NLP

3.2. Feature Extraction

One of the most difficult tasks in sentiment analysis is known as feature extraction. Several techniques are available that are applied to perform feature extraction, such as TF-IDF, N-gram, Doc2Vec, etc. In the proposed work, TF-IDF is applied for feature extraction because it is based on a weight metric that finds the importance of words for the document.

3.2.1. Term Frequency-Inverse Document Frequency (TF-IDF)

TF is used to determine the number of times based on a particular term t that arises in document d. If any term increases several times, then its frequency is also increased. TF is computed by applying the ratio of two terms that are t and d, where t denotes the frequency of a term in a document and d denotes the number of terms in the particular document. TF is responsible for checking the frequency of stop words that occur multiple times [24].

The generalized formula of TF is discussed below:

$$\mathrm{TF}_{(\mathrm{t,doc})}=\frac{\text{number of times term t appears in a text document (doc)}}{\text{total number of terms in a text document (doc)}} \tag{1}$$

The responsibility of IDF is applied to check the importance of the terms that occur in the document. It also gives importance to the words that are occurred several times in the document. The Formula of IDF is described below:

$$TF-IDF_{(t,doc)} = TF_{(t,doc)} \times IDF_{(t)} \tag{2}$$

Where t is denoted as a term, the doc is denoted as a document. The extracted text features representing the text terms are further passed to the next step. For feature extraction, opinions were collated and represented using the tf-idf scheme. The opinions (O_P) has been represented in the form of vectors represented in the given equation:-

$$O_P = (v_{1P,} v_{2p}, v_{3p} \dots \dots, v_{tp}) \tag{3}$$

Each vector dimension belongs to a distinct word (w), or the term used for the respective opinion in the tweets is p. The relation of opinion with the word has been drawn in the form of a matrix given as follows in Table 2:-

Table 2. Opinion (O_P) vs Word (w) matrix

	Word (w) 1	**Word (w) 2**	**………**	**Word (w) n**
O_P **1**	**0**	**0.4**		**0.3**
O_P **2**	**0.7**	**0**		**0**
…				
O_P **n**	**0.2**	**0**		**0.8**

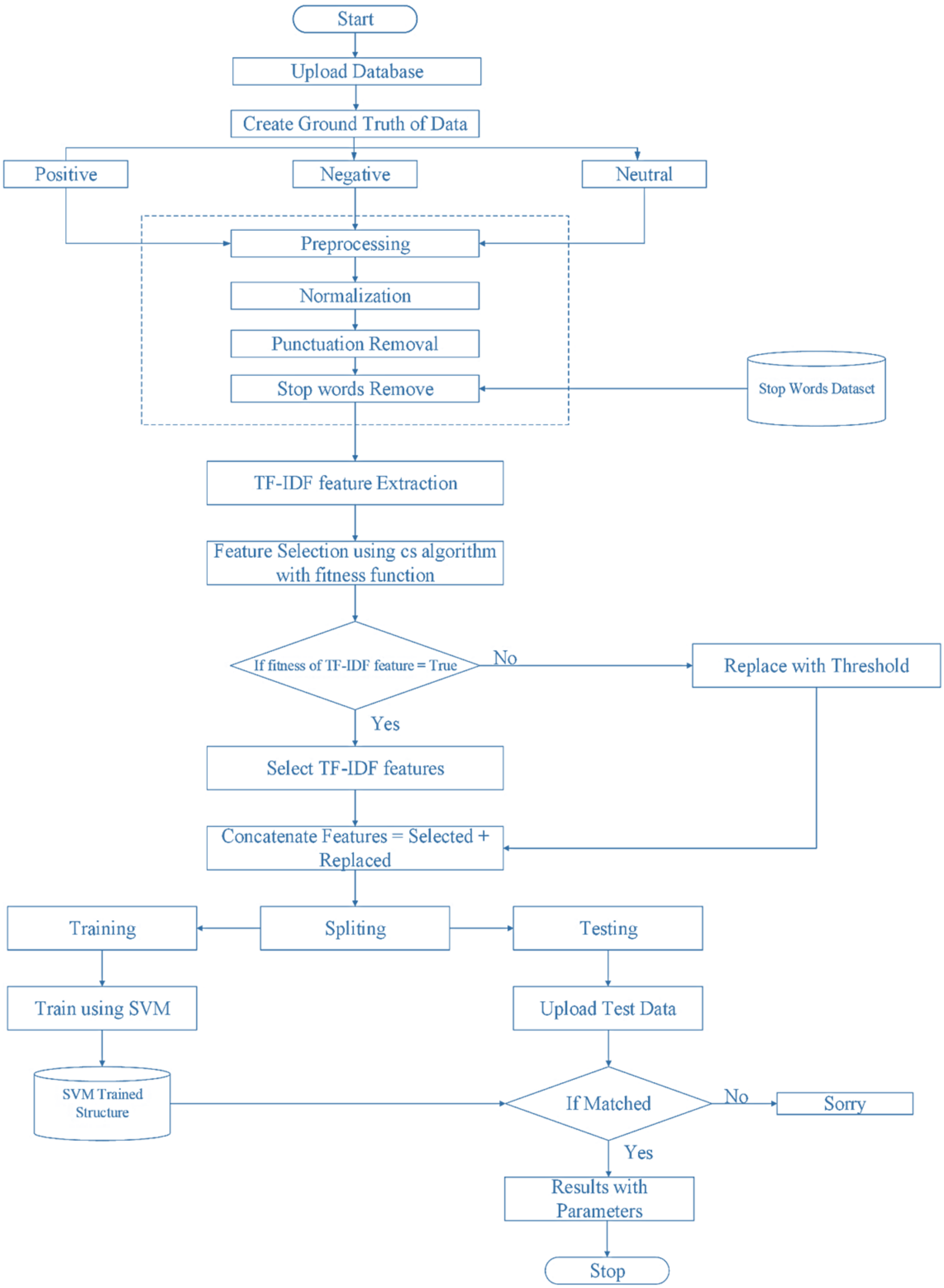

Fig. 2 Work Flow Diagram

Tf-idf is a widely popular term used for the weighting system, and it can be described as follows: a set of documents given as follows,

$$Doc = doc_1, doc_2, \ldots\ldots doc_n \tag{4}$$

With the condition that $d_j \in Doc$, the term frequency represented as $tf(doc, t)$, it is the frequency of term 't' repeated in a doc.

Where, idf(t) represents the document count in which the term repeats only once. However, the inverse of 'tf' in the collating of documents can be computed using equations 5 and 6. Therefore, higher weight has been assigned to the terms that are repeated in a very frequent manner. On the other side, weight decreases using the 'idf' in case the term is repeated many times in a document.

$$idf(t) = \log\left(\frac{|Doc|}{Doc\ F(t)}\right) \tag{5}$$

$$wt.(t) = tf(doc, t) * idf(t) \tag{6}$$

After the development of the matrix model, the cosine similarity model was applied to determine the similarity level between the two O_P such as $O_{P1}\, and\, O_{P2}$. This measure has been widely used in the literature (Orkphol & Yang 2019) and can be used to determine the cosine angle between the two docs as given in equation 7 [25]. The results of the opinion similarity index are given in table 3, which is given as input to the pre-processing.

$$\cos(s_k, s_p) = \frac{(s_k^t, s_p)}{|s_k||s_p|} \tag{7}$$

Table 3. Similarity Matrix for Opinion (O_P)

	Opinion (O_P) 1	O_P**2**	**………**	O_P **n**
O_P **1**	1	0.4		0.3
O_P **2**	0.7	1		0
…				
O_P **n**	0.2	0		1

Similar values are stored and then documented for further processing.

3.3. Feature Selection

Feature extraction is an approach to select a subset of features that differentiate between the various classification categories. The major role of feature selection is to improve computational performance by applying different techniques. It is applied to rank the feature by using metrics and removing all the features rather than top-ranked ones.

3.3.1. Cuckoo Search (CS)

CS is a metaheuristic optimization algorithm that is inspired by the bird cuckoo. It is based on the obligate brood parasitism conduct of cuckoos. Yang and Deb introduced it in the year of 2009 [26]. Nowadays, CS is implemented to solve many complex problems. Pandey *et al.* (2017) proposed a metaheuristic approach based on the CS and K-means for sentiment analysis. The generalized algorithm of CS is discussed below [27]:

Algorithm 1: CUCKOO SEARCH

Selected features = CSA (TF-IDF feature, fitness fun)

1. Start
2. Calculate the size of TF-IDF features
3. R,C = size (TF-IDF features)
4. For i in range(R)
5. For j in range (C)
6. C_{egg} = TF-IDF features(i, j)
7. $T_{Egg} = \frac{\sum_{m=1}^{j} features(i,j)}{j}$
8. fitness fun = $\begin{cases} True; & if\ change\ in\ feature\ is\ minimum\ to\ threshold \\ & False; else \end{cases}$
9. Index = CSA (C_{Egg} , T_{Egg}, fitness fun)
10. End for
11. End for
12. Selected features = tf-idf feature (index)
13. Return: selected features
14. End

The CS algorithm works by analyzing the features extracted using tf-idf features. CS is known for its fast coverage speed while analyzing the possible solutions in the search space. Therefore, it is implemented to replace the not-so-good solution with a potentially better solution. In other words, it CS fitness function is used to identify the optimal solution based on the threshold value. This results in the optimal features selected to represent the textual sentiments. The output of the feature selection phase is passed to the classifier to perform training and classification. According to the proposed CS algorithm, each egg placed in the nest represents a solution and a new possible solution for the foreign cuckoo egg. The main goal of applying this algorithm is to potentially place a better solution by replacing the nest solution. CS generates a new solution ($y_i^{(l+1)}$) for n cuckoo.

$$(y_i^{(l+1)}) = y_i^{(l)} + \forall \otimes levy(f/2) \tag{8}$$

In the given equation 8, l is the step size and $\forall > 0$ for the scaling of step size related to the problem, and it is set to 1 in most of the cases. The concept of Cs is based on the levy flight used to avoid the problem of local optima, and more solution space has been explored using the levy distribution for large steps given as follows:-

$$Levy\left(\frac{f}{2}\right) \rightarrow \sim \cup = S_S^{-\frac{f}{2}}, \left(1 < \frac{f}{2} \leq 3\right) \tag{9}$$

Where f is the frequency and S_s is the size of the steps equivalent to levy steps.

3.4. Training and Classification

In the proposed work, training and classification are performed separately by applying machine learning approaches, namely, Naïve Bayes (NB) Classifier and Support Vector Machine (SVM). The algorithmic structure used during implementation and analysis is discussed individually for NB and SVM.

3.4.1. Naïve Bayes (NB) Classifier

NB classifier is a collection of classification algorithms that belong to the Bayes theorem. It is also known as a family of algorithms in which all algorithms share a common principle. Several authors implement NB to solve different problems due to its simplicity. Goel *et al.* (2016) applied an NB classifier on a real-time Twitter dataset for classification [28]. The NB classifier algorithm is discussed below.

Algorithm 2: NB Algorithm

Classified result = NB (selected features)

1. Start
2. Category record = selected features (1 → 3) // for three classes such as positive, negative, neutral
3. For i in range(category record)
4. If selected features ϵ 1st portion
5. Group (i) = 1
6. Else if selected features ϵ 2nd portion
7. Group (i) = 2
8. Else
9. Group (i) = 3
10. End-if
11. End for
12. Call NB using machine learning
13. Training data = selected feature
14. SA- structure NB = NB(training data, group)
15. Test data = feature of test data
16. Classified result = SA-structure NB (test data)
17. If classified result == 1
18. Return: Positive
19. Else if classified result == 2
20. Return: negative
21. Else
22. Return: Neutral
23. End-if
24. End

The above-discussed algorithm inputs the selected features after CS-based optimization. After training using optimized features, the sentiments are classified into three categories, namely, positive, negative, and neutral. The classification results are then evaluated to identify their performance to classify sentiments into three classes

3.4.2. Support Vector Machine (SVM)

SVM is a machine learning technique that comes under supervised learning. Vapnik first introduces SVM in the year of 1995 [29]. It is applied for classification and regression tasks. SVM provides a facility to classify linear and non-linear datasets. It is the best classification algorithm that provides high dimensions with the datasets. Several researchers have implemented SVM to classify the pattern in the current era. It is concerned with the optimal linear decision surface that deals with the binary classes. The proposed work uses SVM to classify text as positive, negative, and neutral. The SVM algorithm is discussed below:

Algorithm 3: Support Vector Machine

Classified result = SVM (selected features)

1. Start
2. Category record = selected features (1 → 3) // for three classes such as positive, negative, neutral
3. For i in range(category record)
4. If selected features ϵ 1st portion
5. Group (i) = 1
6. Else if selected features ϵ 2nd portion
7. Group (i) = 2
8. Else
9. Group (i) = 3
10. End-if
11. End for
12. Call SVM using machine learning
13. training data = selected feature
14. SA- structure SVM = SVM(training data, group, kernel = RBF)
15. Test data = feature of test data
16. Classified result = sa-structure svm (test data)
17. If classified result == 1
18. Return: Positive
19. Else if classified result == 2
20. Return: negative
21. Else
22. Return: Neutral
23. End-if

Similar to the NB classifier, the SVM distinguishes sentiments into three classes. The RBF kernel trick is implemented to construct a trained SVM structure. Based on the trained SVM structure, the sentiments are classified into three classes, followed by the performance analysis.

4. Results and Discussion

The effectiveness of the proposed work is evaluated using 10000 samples. The performance analysis is performed regarding precision, recall, and classification accuracy.

Table 2. Ordinals of the Implementations

Parameter	Description
Number of Data Samples	100 to 10000
Feature Extraction Technique	TF-IDF
Optimization Technique	CS
Classifiers	NB, SVM
Number of Classes	3
Performance Parameters	Precision, Recall, and Accuracy

The experimental results were performed on the three performance parameters: precision, recall, and accuracy.

- Precision: This parameter quantifies the number of positive class predictions that belong to the positive class.

$$\text{Precision} = \frac{T_P}{T_P+F_P} \tag{3}$$

- Recall: This metric quantifies the number of correct positive generated test data from all positive test data. It indicates missed positive test data.

$$\text{Recall} = \frac{T_P}{T_P+F_N} \tag{4}$$

- Accuracy: The computation of accuracy is done by utilizing True Positive (T_P), True Negative (T_N), False Positive (F_P) and False Negative (F_N).

The effectiveness of the proposed sentiment analysis and opinion mining work is evaluated using different scenarios: Naïve Bayes with and without CS and SVM implemented with and without CS-based optimization of the extracted features. These combinations are implemented separately and analyzed for observed precision, recall, and accuracy of classification into three classes.

Fig. 3 describes the precision analysis performed by applying two classifiers, Naïve Bayes and SVM and both are implemented using CS optimization. The above figure describes four different scenarios with NB, CS-NB, SVM, and CS-SVM. The x-axis defines the number of samples ranging from 100 to 10000 against precision. The precision value of NB and CS-NB is computed as 0.771% and 0.798%. In the second scenario, the precision value of SVM and CS-SVM is computed as 0.852% and 0.987%. The overall precision analysis describes that CS-SVM performed better as compared to CS-NB.

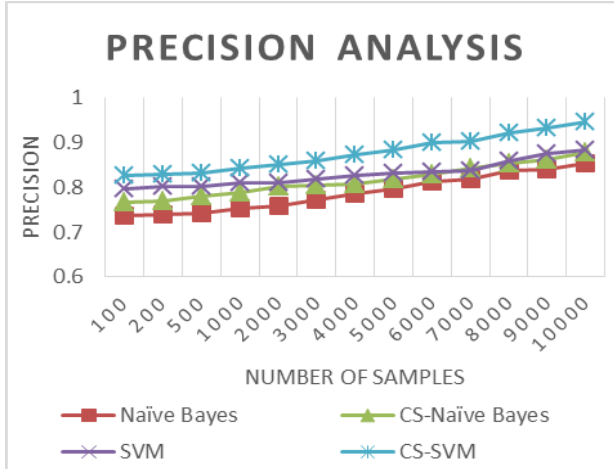

Fig. 3 Precision Analysis

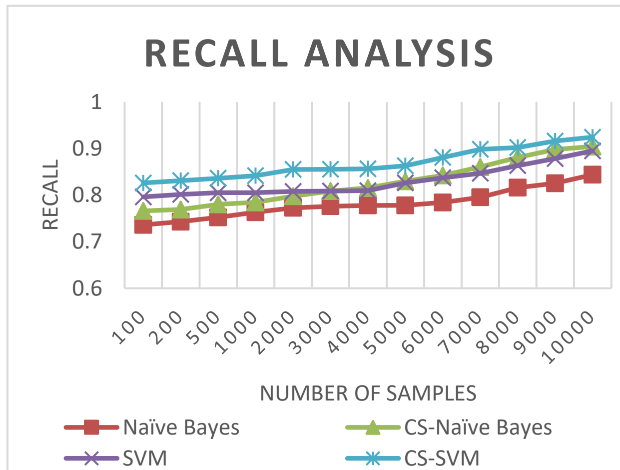

Fig. 4 Recall Analysis

Fig. 4 illustrates the recall analysis that is applied to the two different scenarios by using NB and SVM with the help of CS optimization. The x-axis defines the number of samples against the recall. The NB classifier is implemented in two ways; only NB and CS-NB. The SVM is implemented in two variations, only SVM and CS-SVM. The overall evaluation defines that the CS-SVM performs better results in terms of recall analysis as compared to CS-NB. The recall value of CS-NB and CS-SVM is computed as 0.852% and 0.952%.

Fig. 5 describes the accuracy analysis in four different variations. The NB and SVM classifiers both are implemented with CS and without CS. The graph's x-axis describes the number of samples, and the y-axis defines the accuracy percentage. The result of the accuracy analysis is computed as 79% by using CS-NB, and it is computed as 93.21% by using CS-SVM. The SVM and CS-SVM provide better results as compared to NB and CS-NB.

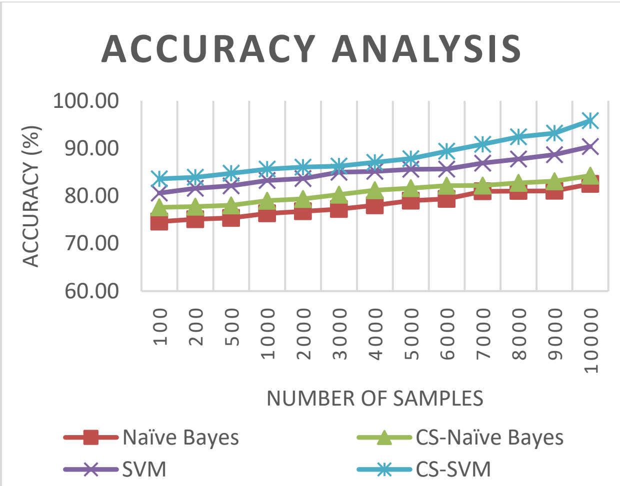

Fig. 5 Accuracy Analysis

Fig. 6 describes the comparative analysis of accuracy based on the proposed work against the existing work. Pandey et al. 2017 applied k-means for sentiment analysis and computed the 78.17% accuracy. Asghar et al. proposed a hybrid sentiment analysis technique and computed 82.37% accuracy. Another author Kermeani et al. [24], applied an SVM classifier and computed the 87.75% accuracy of the work. The proposed work implements CS-SVM, and the overall accuracy is computed at around 93%. The overall evaluation describes that the proposed work provides better results as compared to other existing approaches due to the integration of the unique fitness function of CS. The CS-based optimization capabilities are also enhanced by introducing pre-processing using data normalization at the initial stages.

Table 4 shows the comparative analysis of the proposed technique against the existing work. Sharma et al. 2018 applied SVM with optimization approaches for sentiment analysis and computed the 91.91% accuracy. Jianqiang et al. 2018 proposed a CNN technique for sentiment analysis and computed it the 87.62% accuracy. The proposed work implements CS-SVM, and the overall accuracy is computed at around 93%. Sharma et al. 2018 applied SVM with optimization approaches for sentiment analysis and computed the 96.55% F-measure. Jianqiang et al. 2018 proposed a CNN technique for sentiment analysis and computed the F-measure of 87.50%. In the proposed work, CS-SVM is implemented, and the overall F-measure is computed at around 96.93%. The overall evaluation describes that the proposed work provides better results as compared to other existing approaches[7].

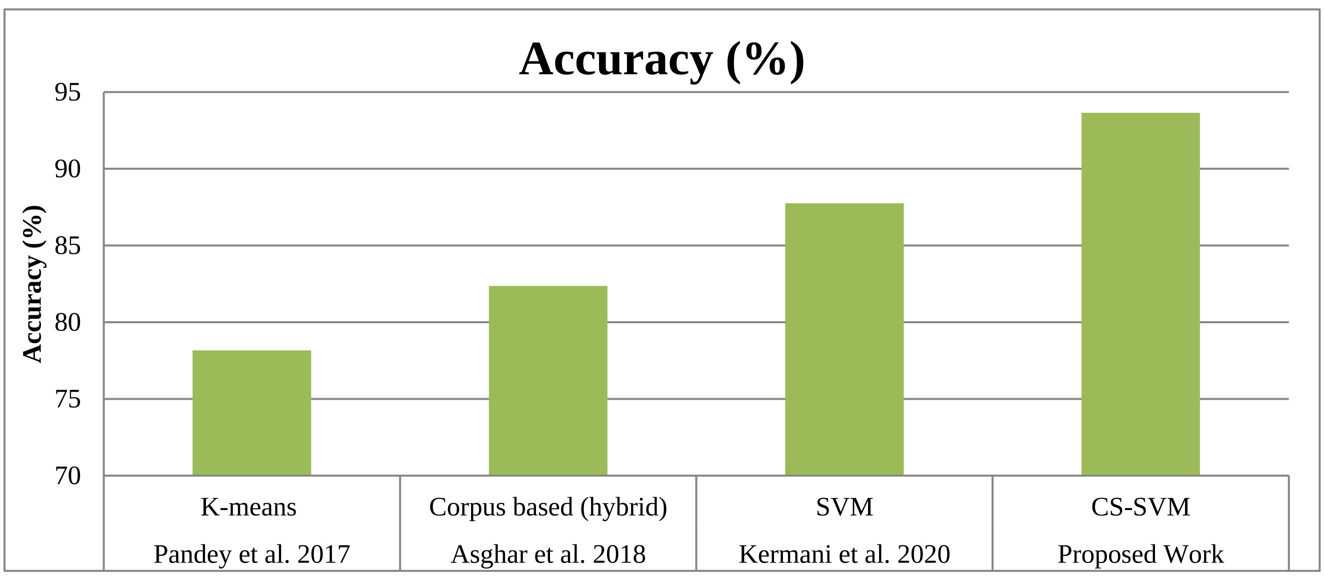

Fig. 6 Comparative analysis of accuracy

Table 4. Comparative analysis based on different performance measures

Performance Measures	Jianqiang et al. 2018 (CNN)	Sharma et al. 2019 (SVM-Hybridization of optimization techniques)	The proposed technique (CS-SVM)
Precision (%)	87.60	98.34	98.72
Recall (%)	87.45	94.83	95.22
F-measure (%)	87.50	96.55	96.93
Accuracy (%)	87.62	91.91	93.21

5. Conclusion and Future Score

The paper presents the research work pertaining to the area of sentiment analysis and opinion mining obtained either from reviews, tweets, forums, discussions, etc., present social media. In the paper, two machine learning classifiers are used to cross-check the effectiveness of CS-optimized opinion mining and classification. The work follows several processing steps, including pre-processing, feature extraction, feature selection, and training and classification. Finally, the evaluation of the designed work is Sentiment analysis for social media performed in terms of precision, recall, and accuracy analysis. The overall results analysis describes that when only NB and SVM are applied separately, the SVM performs better than the NB achieving an average accuracy of 93%. Compared to the existing approaches, the comparative analysis of the proposed work using CS-SVM also symbolizes its success. In the future, the integration of multiclass will be done to improve the classification accuracy of the proposed work.

References

[1] B. Liu, "Sentiment Analysis and Opinion Mining, " *Synthesis Lectures on Human Language Technologies,* vol. 5, no. 1, pp. 1–167, 2012.

[2] A. Buche, D. Chandak, and A. Zadgaonkar, "Opinion Mining and Analysis: A Survey," *Arxiv Prepr. Arxiv1307.3336*, 2013.

[3] Ray P, "Document Level Sentiment Analysis for Product Review Using Dictionary Based Approach," *SSRG International Journal of Computer Science Engeering*, 2017 [Cited 2022 Sep 30]; vol. 4, no. 6, pp. 24–9, 2017.

[4] M. Z. Asghar, F. M. Kundi, S. Ahmad, A. Khan, and F. Khan, "T-Saf: Twitter Sentiment Analysis Framework Using a Hybrid Classification Scheme, *Expert System*, vol. 35, no. 1, pp. E12233, 2018.

[5] Surendiran R, Duraisamy K, "An Approach In Semantic Web Information Retrieval," *SSRG- International Journal of Electronics and Communication Engineering*, [Cited 2022 Sep 30], vol. 1, no. 1. pp. 17–21, 2014.

[6] G. Vinodhini and R. M. Chandrasekaran, "Sentiment Analysis and Opinion Mining: A Survey," *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 2, no. 6, pp. 282–292, 2012.

[7] B. Liu and Others, "Sentiment Analysis and Subjectivity," *Handbook Natural Language Process*, vol. 2, pp. 627–666, 2010.

[8] D. Sharma and M. Sabharwal, "Sentiment Analysis for Social Media Using Svm Classifier of Machine Learning," *The International Journal of Innovative Technology and Exploring Engineering*, vol. 8, no. 9, pp. 39–47, 2019.

[9] Z. Jianqiang, G. Xiaolin, and Z. Xuejun, "Deep Convolution Neural Networks for Twitter Sentiment Analysis," *IEEE Access*, vol. 6, pp. 23253–23260, 2018.

[10] S. Tedmori and A. Awajan, "Sentiment Analysis Main Tasks and Applications: A Survey," *Journal of Information Processing Systems*, vol. 15, no. 3, pp. 500–519, 2019.

[11] G. Saha, S. Roy, and P. Maji, "Sentiment Analysis of Twitter Data Related to Covid-19," *In Impact of Ai and Data Science In Response to Coronavirus Pandemic*, Springer, pp. 169–191, 2021.

[12] A.Kowcika, A. Gupta, K. Sondhi, N. Shivhre, and R. Kumar. "Sentiment Analysis for Social Media." *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 3, no. 7, pp. 216-221, 2013.

[13] 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.

[14] K. Zvarevashe and O. O. Olugbara, A Framework for Sentiment Analysis with Opinion Mining of Hotel Reviews, In *2018 Conference on Information Communications Technology and Society- ICTAS*, pp. 1–4, 2018.

[15] F. Zarisfi Kermani, F. Sadeghi, and E. Eslami Solving the Twitter Sentiment Analysis Problem Based on A Machine Learning-Based Approach, *Evolutionary Intelligence*, vol. 13, no. 3, pp. 381–398, 2020.

[16] S. S. Aljameel *et al*, A Sentiment Analysis Approach to Predict An Individual's Awareness of the Precautionary Procedures to Prevent Covid-19 Outbreaks In Saudi Arabia, *International Journal of Environmental Research and Public Health*, vol. 18, no. 1, pp. 2018, 2021.

[17] G. Angiani *et al*, "A Comparison Between Preprocessing Techniques for Sentiment Analysis in Twitter," 2016.

[18] I. Hemalatha, G. P. S. Varma, and A. Govardhan Preprocessing the Informal Text for Efficient Sentiment Analysis, *International Journal of Emerging Trends in Science and Technology*, vol.1, no. 2, pp. 58–61, 2012.

[19] R. Duwairi and M. El-Orfali, A Study of the Effects of Preprocessing Strategies on Sentiment Analysis for Arabic Text, *Journal of Information Science*, vol. 40, no. 4, pp. 501–513, 2014.

[20] S. Pradha, M. N. Halgamuge, and N. T. Q. Vinh, "Effective Text Data Preprocessing Technique for Sentiment Analysis in Social Media Data," In *2019 11th International Conference on Knowledge and Systems Engineering- KSE*, pp. 1–8, 2019.

[21] Shubhi Kulshrestha, Ankur Goyal, "Cuckoo Search Algorithm and Bf Tree Used for Anomaly Detection in Data Mining, " In*ternational Journal of Computer and Organization Trends*, vol. 9, no. 4 , pp. 11-18, 2019.
Crossref, https://doi.org/10.14445/22492593/IJCOT-V9I4P303.

[22] M. Javed and S. Kamal, "Normalization of Unstructured and Informal Text in Sentiment Analysis," *International Journal of Advanced Computer Science and Applications*, vol. 9, no. 10, 2018.
[23] A. P. Jain and P. Dandannavar, "Application of Machine Learning Techniques to Sentiment Analysis," In *2016 2nd International Conference on Applied and Theoretical Computing and Communication Technology - ICATCCT*, pp. 628–632, 2016.
[24] M. Avinash and E. Sivasankar, "A Study of Feature Extraction Techniques for Sentiment Analysis," *In Emerging Technologies In Data Mining and Information Security*, Springer, pp. 475–486, 2019.
[25] K. Orkphol and W. Yang, "Word Sense Disambiguation Using Cosine Similarity Collaboratewith Word2VEC and Wordnet, *Future Internet*, vol. 11, no. 5, pp. 114, 2019.
[26] X.-S. Yang and S. Deb, "Engineering Optimisation by Cuckoo Search," *Arxiv Prepr. Arxiv1005.2908*, 2010.
[27] M. Elhoseny, H. Elminir, A. Riad, and X. Yuan, "A Secure Data Routing Schema for Wsn Using Elliptic Curve Cryptography and Homomorphic Encryption," *Journal of King Saud University - Computer and Information Sciences*, 2016, Doi: 10.1016/J.Jksuci.2015.11.001.
[28] V. Vapnik, "The Nature of Statistical Learning Theory," *Springer Science \& Business Media*, 1999.
[29] A. Goel, J. Gautam & S. Kumar, "Real Time Sentiment Analysis of Tweets Using Naive Bayes," I*n 2016 2nd International Conference on Next Generation Computing Technologies NGCT*, pp. 257-261, 2016.
[30] A. Krouska, C. Troussas, and M. Virvou, "The Effect of Preprocessing Techniques on Twitter Sentiment Analysis," *In 2016 7th International Conference on Information, Intelligence, Systems & Applications - IISA*, pp. 1–5, 2016.
[31] uska, C. Troussas, and M. Virvou, "The Effect of Preprocessing Techniques on Twitter Sentiment Analysis," *In 2016 7th International Conference on Information, Intelligence, Systems & Applications - IISA*, pp. 1–5, 2016.