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

Improving Opinion Mining Accuracy with Dragonfly Algorithm-Based Hybrid Classification

Mikanshu Rani*[1], Jaswinder Singh[2]

Submitted: 28/03/2024 **Revised:** 21/04/2024 **Accepted:** 29/04/2024

Abstract: The paper proposes the Dragonfly + Hybrid Classifier, a novel approach designed to enhance opinion mining across diverse datasets. Leveraging the Dragonfly algorithm for feature set selection and combining it with a hybrid classification method, this innovative approach offers the potential for more accurate and reliable predictions. On the Twitter Sentiment dataset, notorious for its dynamic and noisy nature, the Dragonfly + Hybrid Classifier excels with an average precision of approximately 0.93498, recall of approximately 0.92965, and an F-measure of approximately 0.93208, alongside an average accuracy of around 96.134%. Within the Movie Review dataset, where opinions are nuanced and context-dependent, the Dragonfly + Hybrid Classifier secures an impressive average precision of approximately 0.91348, coupled with an average recall of approximately 0.9189, achieving an F-measure of approximately 0.91582 and maintaining an average accuracy of around 94.98%. In the context of the Depression dataset, where sensitivity and accuracy are paramount, the Dragonfly + Hybrid Classifier excels with an average precision of approximately 0.9627, an average recall of approximately 0.966, an F-measure of approximately 0.9643, and an average accuracy of around 94.62%. These findings collectively affirm the Dragonfly + Hybrid Classifier as a potent tool for opinion analysis across diverse domains, positioning it as a valuable asset in field of opinion mining and analysis applications, particularly in domains where opinion understanding is paramount.

Keywords: *Machine Learning, Dragonfly, Hybrid, TF-IDF, Datasets.*

1. Introduction

Opinion Mining, also known as Sentiment Analysis, is a field of Natural Language Processing (NLP) and data science that focuses on extracting and analysing subjective information from text data, such as reviews, social media posts, news articles, and more. It involves the use of computational techniques to determine the sentiment or opinion expressed in these texts, which can be positive, negative, or neutral. Opinion Mining has gained immense relevance in the modern world due to its potential to uncover valuable insights, influence decision-making processes, and shape various aspects of society. Opinion Mining has become increasingly relevant in the modern world for several reasons [1].

a. Big Data and Social Media: Opinion Mining techniques are crucial for organizations and individuals to make sense of this vast amount of information and extract valuable insights from it [2].
b. Customer Feedback and Brand Management: Businesses can use Opinion Mining to analyse customer reviews, feedback, and social media mentions to understand how their products or services are perceived by the public [3].
c. Financial Markets: Opinion Mining is increasingly used in financial markets to analyse news articles, social media posts, and other textual data to predict market trends and opinions [4]. Traders and investors use these insights to make informed decisions.
d. Healthcare: In the healthcare industry, Opinion Mining can be applied to patient reviews and medical records to understand patient satisfaction, identify potential issues, and improve the quality of care.
e. Disaster Response and Crisis Management: During emergencies, monitoring public opinion through social media can help authorities respond more effectively.
f. Academic and Scientific Research: Researchers use Opinion Mining to analyze academic papers, survey responses, and other textual data to gain insights into public perception, trends in scientific literature, and emerging research topics [5].

Machine learning (ML) in opinion mining has undergone a remarkable transformation driven by the proliferation of digital platforms and the unprecedented volume of data they generate. This enabled the development of more sophisticated opinion mining techniques capable of handling vast and diverse text datasets. Opinion mining models, including traditional classifiers and deep learning architectures, have emerged as powerful tools for recognizing and categorizing opinions within text data [6]. An overview of generalized ML architecture is presented in Fig. 1 depicting various steps and algorithms involved in training and classification process. They not only classify opinions as positive, negative, or neutral but also perform fine-grained analysis and aspect-based opinion analysis, providing deeper insights into subjective information. However, ML for opinion mining faces various challenges [7]. The quality of data from digital platforms can be compromised by noise, slang, and contextual ambiguity, posing difficulties for accurate opinion interpretation [8]. Domain-specific variations in language and opinion expressions further

1 Ph. D Scholar, Deptt. of Computer Science & Engineering, Guru Jambheshwar University of Science & Technology, Hisar, India
2 Professor, Deptt. of Computer Science & Engineering, Guru Jambheshwar University of Science & Technology, Hisar, India
1mikanshu6406@gmail.com,
2jaswinder_singh_2k@rediffmail.com
** Corresponding Author Email: mikanshu6406@gmail.com*

challenge model generalization. Imbalanced datasets, ethical concerns regarding bias in models, and the need for real-time processing are additional issues that necessitate ongoing research and development in the field [9]. As opinion mining evolves, addressing these challenges is crucial in harnessing the full potential of ML in extracting valuable insights [10].

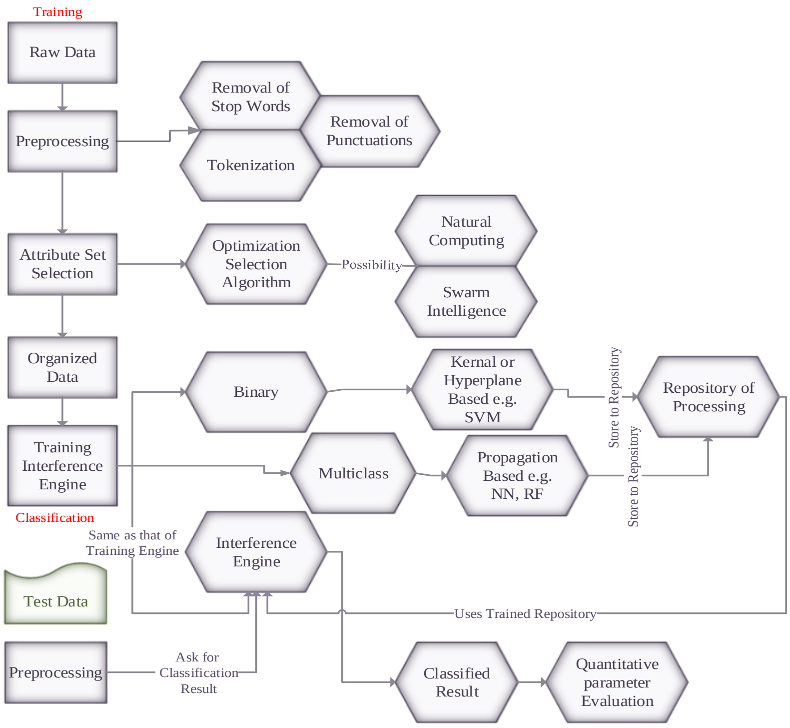


Fig. 1 General Machine Learning Architecture

Considerable research efforts have been dedicated to the field of opinion mining in the past. In this context, the proposed work represents a significant advancement by introducing a novel hybrid model that involves feature set selection and classification techniques [11]. Specifically, this approach leverages the concept of dragonfly algorithm for feature set selection, ensuring that the most relevant and informative features are retained for opinion mining and analysis [12, 13]. Feature selection is a critical step in opinion-based analysis as it helps streamline the data and enhance the model's efficiency. The contributions are listed as follows.

a. Introduction to the Hybrid Model: This paper describes a novel hybrid model for opinion mining and analysis that involves feature set selection and classification approaches.
b. Feature Set Selection via Improved Dragonfly method: The model selects features using an innovative dragonfly method that retains the most important features from the textual data.
c. Deep Neural Network (DNN) Integration: To capture intricate patterns and correlations in text data, the hybrid model includes a DNN.
d. Multiple Classifier Integration: In addition to DNN, the model incorporates SVM, resulting in a robust and adaptive solution for a wide range of data features and complexity.
e. Evaluation on Three Different Datasets: The hybrid model was carefully tested on three different datasets to demonstrate its generalizability and flexibility across multiple domains.
f. Advancement in Opinion Mining: This work makes substantial contributions to opinion mining by combining feature selection, various classifiers, and deep learning, providing a promising method for increasing accuracy and applicability.

The rest of the paper is organised in the following manner. Section 2 illustrates the related work whereas the proposed work is described in section 3. The evaluation of the results have been discussed in section 4 and the paper is concluded in section 5.

2. Related Work

The last decade researches in the field of opinion mining have been found attracted and influenced by the optimization approaches. Several researchers have presented architectures to resolve either feature extraction or classification stage using different techniques and algorithms. In similar context, Marie-Sainte and Alalyani, (2020) [1] presented an innovative application of the Firefly Algorithm in the context of Arabic text classification. This approach filled a crucial gap in the field by introducing a feature selection technique tailored to Arabic sentiment analysis, thus contributing significantly to advancements in Arabic NLP. Elangovan and Subedha (2020) [6] proposed a holistic model for sentiment analysis, combining the Firefly algorithm, Levy flight, and Multilayer Perceptron. This work represents a valuable contribution to the development of sophisticated sentiment analysis tools. Chantar et al. (2020) [9] proposed binary Grey Wolf Optimizer (GWO) with elite-based crossover for feature selection in Arabic text classification, this study contributed by enhancing the GWO for feature selection. The primary contribution lies in optimizing the GWO technique to improve its effectiveness in opinion analysis tasks, particularly for the Arabic language. By introducing this enhanced GWO variant, the paper made progress in the field of optimization for text classification. Asgarnezhad et al. (2021) [14], the author introduced the concept of Multi-Objective Gray Wolf Optimization (MOGW) for feature selection in text classification. This research made a noteworthy contribution by broadening the repertoire of optimization techniques available for text classification tasks, offering potential benefits in terms of classification accuracy and efficiency. Alarifi et al. (2020) [15] adopted a big data approach to opinion analysis, combining greedy feature selection with Cat Swarm Optimization-based Long Short-Term Memory (LSTM) neural networks. The significant contribution here lies in addressing the challenges posed by extensive datasets. This approach represents a substantial step towards handling the increasing volumes of textual data generated in the digital age. Tubishat et al. (2019) [16] focused on Arabic sentiment analysis and opinion mining, this research enhanced the Whale Optimization Algorithm (WOA) for feature selection. The key contribution is in the improvements made to the WOA, specifically tailored to optimize feature selection for opinion mining tasks. This contribution improved the applicability and effectiveness of WOA, particularly for Arabic language data.

3. Proposed Work

The methodology of the proposed work involves several steps, including, stop word removal, stemming, feature extraction using TF-IDF, cosine similarity, Euclidean distance. In the later stages dragonfly optimization approach with hybrid classifier are also involved.

3.1. Significance and Work Architecture

The proposed work is structured into two distinct segments, each dedicated to enhancing specific aspects of opinion mining and analysis. The first segment focuses on feature set selection, aiming to improve the efficiency and effectiveness of this critical step. In this phase, the proposed work introduces novel advancements to the Dragonfly Algorithm, tailoring it for feature set selection. The primary objective is to refine the feature set selection process, ensuring that the most relevant and informative features are chosen from the textual data. This enhancement aims to elevate the quality of input features for opinion mining, subsequently impacting the overall accuracy of the classification process.

In the second segment of the proposed work, attention is directed towards optimizing the classification phase. Here, a hybrid classifier is formulated, integrating various classification techniques to boost the overall performance of opinion mining. The hybrid classifier combines the strengths of different classification algorithms, creating a synergy that leverages their individual capabilities. This approach intends to enhance accuracy and robustness, especially when confronted with diverse and complex textual data. By creating this hybrid classifier, the proposed work aims to provide a comprehensive solution that addresses the challenges of opinion mining and analysis across various domains and data sources. Fig. 2 shows how these two segments of the proposed work synergize to create a holistic opinion mining framework. The refined feature set selection process optimizes the input data quality, while the hybrid classifier elevates the classification accuracy, collectively contributing to a more robust and efficient opinion mining system.

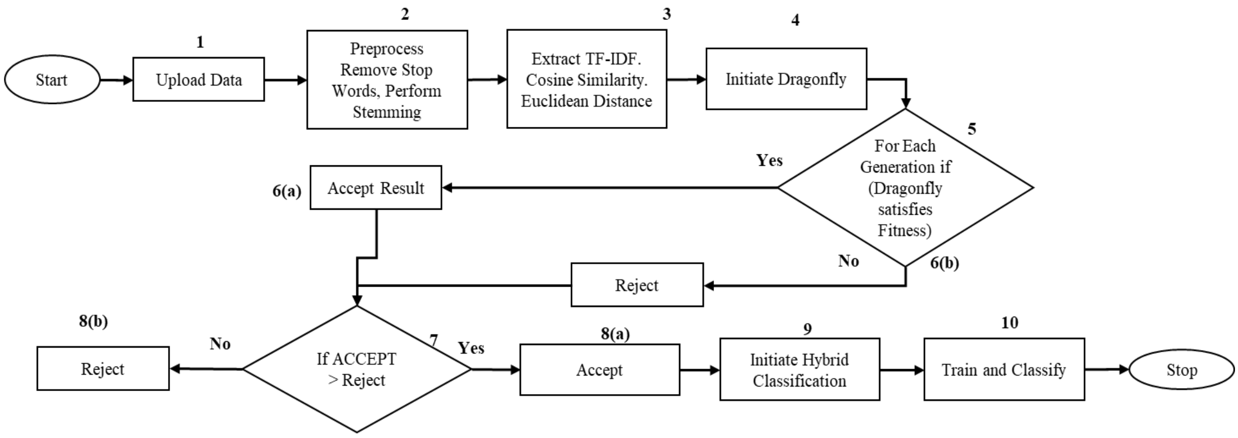


Fig. 2 Overall Proposed Work

3.2. Datasets

The work is explored using three datasets namely, twitter, movie and depression datasets.

3.2.1. Twitter dataset (Sentiment140):

The dataset includes 100,000 tweets, each labeled with sentiment polarity and divided into three categories: 0 for negative emotion, 1 for neutral mood, and 2 for positive sentiment. The dataset is available on Kaggle and can be viewed via the public repository [17]. The Sentiment140 dataset is commonly used for sentiment analysis tasks. This dataset is utilized by researchers and data scientists for sentiment analysis experiments as well as machine learning tasks related to sentiment categorization and opinion mining.

3.2.2. Movie Reviews dataset (NLTK Movie Review):

The dataset includes 60,000 movie reviews, each with a binary sentiment: positive or negative. The NLTK Movie Review dataset, which is available for download from Kaggle's online repository [18] is commonly used for sentiment analysis, text classification, and sentiment classification tasks. Each review is classified as favorable or negative, making this dataset useful for NLP research and applications. It is frequently used by researchers and machine learning practitioners to create and assess sentiment analysis models, as well as to perform text classification and sentiment prediction tasks, notably with regard to movie reviews and general text data.

3.2.3. Depression dataset (Sentimental Analysis for Tweets):

The dataset contains 10,000 tweets, each of which is labeled to identify whether it exhibits depression-related attitudes (1) or not

(0). The Depression dataset, which can be accessed through Kaggle's online repository [19], is designed for sentiment analysis of tweets about depression and mental health. It is useful for researching and analyzing mental health sentiments, as well as developing sentiment analysis models aimed at detecting and raising awareness of depression. Researchers, mental health practitioners, and data scientists can use this dataset to better understand sentiment patterns connected to depression and mental health difficulties, as well as do sentiment analysis and categorization to get insights into public sentiment on these topics.

3.3. Feature Extraction and Selection (Phase 1)

In the context of the proposed work, the first essential step is the calculation of TF-IDF (Term Frequency-Inverse Document Frequency) from the dataset. TF-IDF is a fundamental technique in natural language processing and information retrieval for assessing the importance of terms within a document or a corpus [20]. The calculation involves two main components:

1. **Term Frequency (TF):** This represents the frequency of a term (word) within a document. It is calculated as follows:

$$\mathrm{TF}(t,d) = \frac{f_{t,d}}{|d|} \tag{1}$$

Where TF (t, d) is the Term Frequency of term 't' in document 'd'. $f_{(t,d)}$ is the frequency of term 't' in document 'd'. |d| is the total number of terms in document 'd'.

2. **Inverse Document Frequency (IDF):** This quantifies the importance of a term across the entire corpus. It is calculated using the formula:

$$\mathrm{IDF}(t,D) = \log\left(\frac{N}{\{d \in D: t \in d\}}\right) \tag{2}$$

Where IDF (t, D) is the Inverse Document Frequency of term 't' in the corpus 'D'. N is the total number of documents in the corpus. | {d ∈ D: t ∈ d} | represents the number of documents containing the term 't'. Once TF-IDF scores have been computed for all terms in the dataset, they serve as crucial features for subsequent analysis, allowing the model to capture the importance of individual terms in distinguishing between categories. To elevate the performance of the feature set selection, two additional computation measures namely Cosine Similarity and Euclidean distance of each document has been incorporated.

3. **Cosine Similarity** is a metric used to measure the similarity between two non-zero vectors in an n-dimensional space [21]. In the context of feature set selection, it is often employed to assess the similarity between documents represented as vectors in a high-dimensional space, such as TF-IDF vectors. The formula for calculating the Cosine Similarity between two vectors A and B is as follows:

$$\text{Cosine Similarity}(A,B) = \frac{(A \cdot B)}{(||A|| * ||B||)} \tag{3}$$

Where A · B represents the dot product of TF-IDF content values A and B. ||A|| and ||B|| denote the Euclidean norms (magnitudes) of TF-IDF vectors A and B, respectively.

4. **Euclidean distance:** A key idea in geometry and mathematics, especially when it comes to vector spaces, is the Euclidean distance. In Euclidean space, which is a space that complies with Euclidean geometry, it calculates the straight-line distance between two points. The Euclidean distance between A and B in an n-dimensional space is computed using the following formula:

$$\text{Euclidean Distance}(A,B) = \mathrm{sqrt}(\Sigma\,(A_i - B_i)^2) \tag{4}$$

Now, transitioning to the application of the Dragonfly algorithm in proposed work. It serves as an important optimization approach for feature set selection that offers refined feature set for the classification models. Dragonfly draws inspiration from the collective behaviour of dragonflies in search of prey, making it a robust and adaptable optimization algorithm. Its utility lies in its ability to efficiently explore the parameter space of classification models, aiming to maximize classification accuracy. In the context of the proposed work, the primary contribution of this approach is enhancing the models' ability to differentiate states based on textual features. The dragonfly algorithm consists of two main phases, namely, exploitation and exploration. It optimizes a set of solutions (represented as "dragons") by mimicking the behaviour of dragonflies. The goal is to select a subset of features that maximizes a fitness function. The processes involved in the two phases are elaborated below.

Exploitation Phase:

In the exploitation phase, the algorithm focuses on exploiting the local search space to improve the solutions. Here's how it works:

1. For each dragon (solution) in the population:
   - Identify the group to which the dragon belongs.
   - Select other dragons from the same group for potential pairing (exploitation).
2. Perform a specified number of iterations (max_flight) to explore potential pairings:
   - Randomly select a set of dragons from the same group.
   - Calculate the alignment and cohesion of the selected dragons.
   - Determine if the selected dragons provide a reward based on alignment and cohesion criteria.
3. Compute a reward for each iteration based on alignment and cohesion criteria.
4. Calculate the mean reward across all iterations for each dragon.
5. If the mean reward is greater than or equal to a predefined threshold (e.g., 8), accept the dragon as a solution, add it to the accepted set, and record its label.

Exploration Phase:

In the exploration phase, the algorithm explores the search space more broadly:

1. Similar to the exploitation phase, select dragons from the same group for potential pairing (exploration).
2. Perform a specified number of iterations (max_flight) to explore potential pairings:
   - Randomly select a set of dragons from the same group.
   - Calculate the alignment and cohesion of the selected dragons.
   - Determine if the selected dragons provide a reward based on alignment and cohesion criteria.
3. Compute a reward for each iteration based on alignment and cohesion criteria.
4. Calculate the mean reward across all iterations for each dragon.
5. If the mean reward is greater than or equal to a predefined threshold (e.g., 8), accept the dragon as a solution, add it to the accepted set, and record its label.

3.4. Training and Classification

The Deep Neural Network (DNN) is a powerful and versatile machine learning model commonly used for classification tasks. It consists of multiple layers of neurons that learn to extract features from the input data and make predictions. DNNs are known for their ability to capture complex patterns and relationships in data, especially when provided with large amounts of training data. The Support Vector Machine (SVM) is another popular machine learning algorithm for classification that finds a hyperplane that best separates data points belonging to different classes while maximizing the margin between them [22]. SVMs are known for their robustness and ability to handle non-linear data using kernel functions. The proposed work combines both these algorithms to form a hybrid classifier with the following conditions and rationale.

3.4.1. Condition

If the weights assigned to the two classes by the DNN are not significantly different, you switch to using SVM as a binary classifier.

3.4.2. Rationale

The decision to switch to SVM is based on the idea that if the DNN is uncertain or assigns similar weights to both classes, it may not be confident in its classification. In such cases, using a simpler and more traditional classifier like SVM can provide a more conservative and reliable classification.

Algorithm 2 Hybrid Classifier Algorithm

Input: Training data X, Labels y
Output: Predicted class for new data X_{new}
Train the Deep Neural Network (DNN) on the training data X and labels y
Define the threshold θ
DNN Classification:
$w_1, w_2 \leftarrow$ Weights assigned to Class 1 and Class 2 by DNN
$\Delta w \leftarrow |w_1 - w_2|$
if $\Delta w \leq \theta$ **then**
    Switch to SVM:
    Train the Support Vector Machine (SVM) on the training data X and labels y
    Predict the class of X_{new} using the trained SVM
else
    Predict the class of X_{new} using the trained DNN
end if
Return Predicted class for X_{new}

If we denote the weights assigned to the two classes by the DNN as follows:

- Let w1 represent the weight assigned to Class 1.
- Let w2 represent the weight assigned to Class 2.

The condition to switch to using SVM as a binary classifier can be mathematically expressed as:

$$If \;\; | \, w1 - w2 \, | \leq threshold \tag{5}$$

Benefits of the Hybrid Approach

- Robustness: The hybrid approach combines the strengths of both DNN and SVM, making the classification process more robust.
- Confidence-Based Classification: It allows the system to make more confident decisions when DNN produces clear weight differences and rely on SVM for uncertain cases.
- Improved Performance: By leveraging both classifiers, you may achieve better overall classification performance, especially when handling complex or ambiguous data.

4. Results and Discussion

In the results section, the comprehensive evaluation of the proposed work is presented, shedding light on key performance metrics such as precision, recall, F-measure, and accuracy. These metrics serve as essential indicators of the effectiveness of the approach in opinion mining. Among the contenders are well-established methods such as DNN, Naive Bayes, and Random Forest, all of which represent distinct approaches to opinion mining. This section serves as a crucial juncture where the empirical evidence is presented and discussed, shedding light on the performance of the method and its competitive standing in the field of opinion mining. It allows for drawing insightful conclusions and gaining a deeper understanding of how the approach fares in the context of opinion classification.

The performance of the proposed Dragonfly+Hybrid Classifier applied to the Twitter dataset, is presented at the initial stance with 100,000 records. This evaluation encompasses essential performance metrics, including precision, recall, and F-measure, to gauge the classifier's effectiveness in opinion classification on Twitter data.

- Precision, a critical metric, assesses the classifier's accuracy in predicting positive sentiment instances computed as eq. 6.

$$\text{Precision} = \frac{\text{Cp}}{Cp+Fp} \tag{6}$$

Here, Cp represents correctly predicted positive sentiments (True Positives), while Fp represents instances where the classifier incorrectly predicted positive sentiments (False Positives).

- Recall is another crucial metric, offering insights into the classifier's ability to capture actual positive sentiment instances. It is calculated as follows:

$$\text{Recall} = \frac{\text{Cp}}{\text{Fn+Cp}} \tag{7}$$

In this equation, Fn accounts for instances where the classifier missed actual positive sentiments (False Negatives).

- F-measure is a comprehensive metric that balances precision and recall, providing a holistic assessment of the classifier's performance. It is computed as:

$$\text{F} - \text{measure} = 2 \times \frac{\text{Precision}*\text{Recall}}{\text{Precision+Recall}} \tag{8}$$

This formula considers both the classifier's precision in identifying positive sentiments and its ability to capture a substantial portion of the actual positive sentiments.

- Accuracy calculates the percentage of positive and negative cases in the dataset that were correctly classified. The following formula defines accuracy mathematically:

$$\text{Accuracy} = \frac{(\text{C}_\text{p} + \text{Cn})}{\text{Total instances}} \tag{9}$$

Total Instances is the total number of instances in the dataset. Cp are instances correctly predicted as positive sentiments; and Cn are instances correctly predicted as negative sentiments. The performance analysis using twitter dataset is summarized in table 1, table 2, table 3 and table 4 for precision, recall, f-measure and accuracy values, respectively with graphical analysis in Fig. 3.

Table 1. Precision for Twitter

'Total number of Samples'	Proposed 'Dragonfly+Hybrid'	DNN'	Naive Bayes'	Random Forest'	Decision Tree'	Elangovan et al.'	P Mudgil et al.'
20000	0.91967037	0.85656843	0.78949711	0.8864647	0.82699406	0.89623808	0.84977975
30000	0.94727324	0.89231654	0.88156339	0.83784049	0.88029764	0.87890304	0.91819081
40000	0.95168988	0.9132804	0.91064851	0.8541374	0.85969343	0.8911827	0.89093842
50000	0.94251128	0.84892772	0.84891855	0.82475625	0.86019482	0.91755555	0.86278929
60000	0.94475219	0.86767657	0.80393149	0.92920665	0.88121199	0.88543154	0.86204227
70000	0.94873864	0.87326494	0.89274556	0.87472956	0.86533141	0.87711521	0.89001016
80000	0.92762952	0.86608519	0.86858922	0.8132916	0.87319859	0.85262937	0.90959483
90000	0.92292958	0.80754023	0.80359749	0.80051021	0.89526068	0.84673401	0.87856826
100000	0.90954048	0.88365808	0.9020403	0.78271294	0.79829496	0.90949297	0.8285045

Table 2. Recall for Twitter

'Total number of Samples'	Proposed 'Dragonfly+Hybrid'	DNN'	Naive Bayes'	Random Forest'	Decision Tree'	Elangovan et al.'	P Mudgil et al.'
20000	0.93080596	0.80858579	0.84176845	0.83640516	0.83374982	0.90995218	0.90194947
30000	0.90047007	0.77674667	0.84315303	0.85691364	0.89563958	0.85783408	0.85124585
40000	0.91699947	0.81170501	0.90446682	0.84181563	0.82791972	0.83032706	0.89750337
50000	0.93150964	0.82388038	0.87336128	0.89567805	0.85595405	0.90188073	0.87738924
60000	0.93813013	0.83666479	0.90735806	0.92918158	0.88665265	0.86570832	0.92191685
70000	0.9065674	0.78946521	0.81526276	0.85928366	0.87091954	0.88455727	0.87725759
80000	0.94783911	0.94505565	0.90941318	0.88671466	0.83848961	0.87626995	0.88041301
90000	0.94698851	0.83777408	0.80637887	0.86346042	0.88918127	0.87217991	0.9393221
100000	0.94750566	0.87715254	0.8265974	0.84849124	0.82257506	0.86550486	0.90250747

Table 3. F-Measure for Twitter

'Total number of Samples'	Proposed 'Dragonfly+Hybrid'	DNN'	Naive Bayes'	Random Forest'	Decision Tree'	Elangovan et al.'	P Mudgil et al.'
20000	0.92520466	0.83188578	0.8147953	0.86070767	0.8303582	0.90304306	0.87508776
30000	0.9232789	0.83053044	0.8619305	0.84726974	0.88790234	0.86824076	0.88345193
40000	0.93402268	0.85950208	0.90754714	0.84793175	0.84350746	0.85967925	0.89420885
50000	0.93697816	0.83621653	0.86096647	0.85875534	0.85806919	0.90965062	0.87002802
60000	0.94142951	0.85188854	0.85251934	0.92919411	0.88392395	0.87545886	0.89097478
70000	0.92717375	0.82925337	0.85224668	0.86693782	0.86811649	0.88082052	0.88358786
80000	0.93762543	0.90384876	0.88853253	0.84841756	0.85549219	0.86428803	0.89476605
90000	0.93480427	0.82237937	0.80498578	0.83079457	0.89221062	0.85926862	0.90792998
100000	0.92813499	0.88039329	0.86267257	0.81427584	0.81025316	0.88695386	0.86392413

Table 4. Accuracy for Twitter

'Total number of Samples'	Proposed 'Dragonfly+Hybrid'	DNN'	Naive Bayes'	Random Forest'	Decision Tree'	Elangovan et al.'	P Mudgil et al.'
20000	95.6899992	83.6263648	86.7038576	82.023452	80.4759494	89.6748291	93.4636376
30000	96.2151934	80.0080574	83.2264146	85.4931751	81.2311209	86.0690484	89.3422168
40000	98.4219704	81.3206338	84.0334657	82.9801019	84.7531173	86.4625445	88.8357737
50000	95.2110785	82.4515261	85.8935041	84.2942252	86.0895872	85.8345018	91.3562426
60000	96.5311804	83.6574671	80.7405879	82.83206	85.3445325	91.529975	88.9721194
70000	95.6778308	83.3939733	85.1888056	81.0499808	80.3111786	88.0968162	93.1543805
80000	96.928332	82.3160052	81.3793766	86.2143967	82.7775406	87.6417688	94.6451274
90000	95.0705149	81.1707748	83.6009642	81.0832664	85.24094	87.2296697	88.7819402
100000	95.4662391	81.0332576	83.9239167	84.0795368	86.9223813	89.2879653	91.3566637

In terms of average Precision, the "Proposed Dragonfly + Hybrid" model stands out with an impressive average Precision of 0.934. This result indicates that this proposed model excels in correctly classifying positive instances while minimizing false positives. Compared to other models like "DNN" (0.867), "Naive Bayes" (0.855), "Random Forest" (0.844), and "DT" (0.860), the proposed model demonstrates a significantly higher Precision score. When it comes to average Recall, "P Mudgil et al." leads the pack with an average Recall value of 0.894. In contrast, the "Proposed Dragonfly + Hybrid" model achieves an average Recall of 0.929, demonstrating its capability to capture a substantial portion of true positives while maintaining a strong balance with Precision. The "Proposed Dragonfly + Hybrid" model achieves an outstanding average F-measure of 0.932, which is significantly higher than the other models. This result underscores the model's exceptional overall performance. "P Mudgil et al." also achieves a relatively high F-measure of 0.884, indicating its ability to strike a balance between Precision and Recall. However, the "Proposed Dragonfly + Hybrid" model still outperforms it. In the realm of accuracy, the "Proposed Dragonfly + Hybrid" model distinguishes itself with an impressive accuracy score of 96.13%.In comparison to other models, such as "DNN" (82.10%), "Naive Bayes" (83.85%), "Random Forest" (83.33%), and "DT" (83.68%), the "Proposed Dragonfly + Hybrid" model demonstrates a significantly higher level of overall correctness. This robust performance makes it a superior choice for applications where exactness in predicting outcomes across the entire dataset is paramount. Furthermore, for the second dataset viz. the movie review dataset, the dataset based analysis have been conducted.

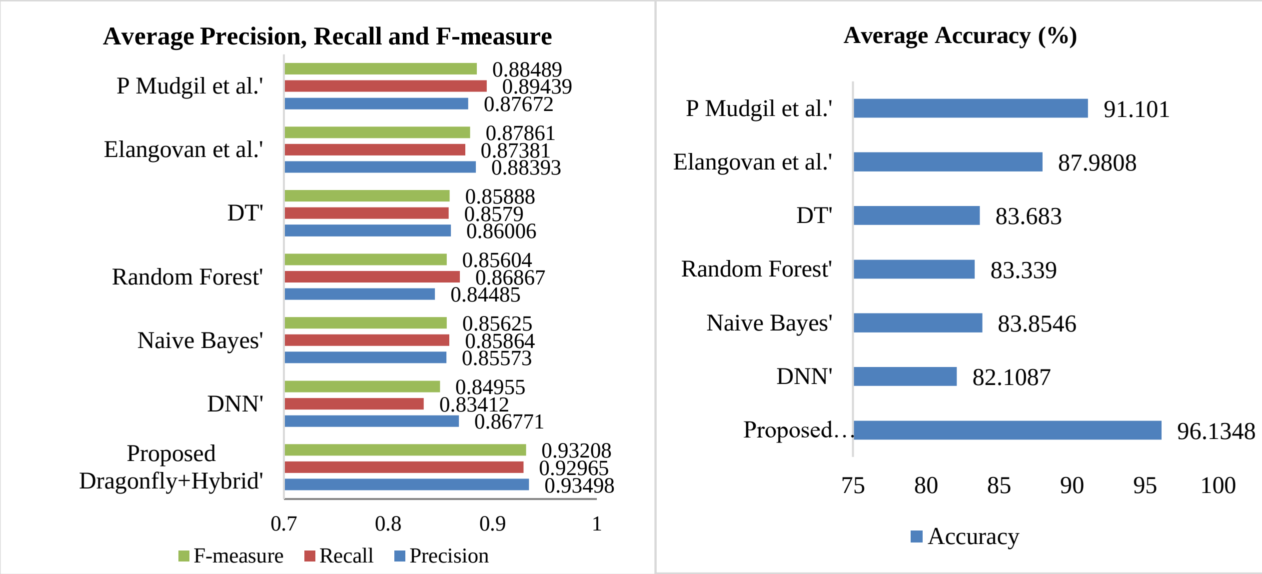


Fig. 3. Average Comparison for Twitter Dataset

Table 5. Precision for Movie Review Dataset

'Total number of samples'	Proposed 'Dragonfly+Hybrid'	DNN'	Naive Bayes'	Random Forest'	Decision Tree'	Elangovan et al.'	P Mudgil et al.'
10000	0.89846017	0.78459979	0.8135714	0.84961833	0.87251573	0.88529639	0.84709717
15000	0.93866141	0.87597997	0.81855994	0.90931022	0.87994407	0.93095771	0.92459891
20000	0.89364545	0.79474013	0.76839142	0.78834377	0.8402158	0.80971948	0.86686802
25000	0.89176597	0.77751738	0.88451063	0.84751267	0.75955035	0.80263471	0.85445841
30000	0.89701551	0.88938576	0.87337883	0.80614005	0.88302051	0.84661567	0.87280118
35000	0.95264592	0.94496458	0.82444007	0.90872414	0.81040782	0.9467352	0.90931145
40000	0.89629242	0.7746026	0.78526748	0.79595516	0.80649519	0.83088911	0.87306237
45000	0.91775646	0.87128349	0.9033275	0.85095809	0.81666451	0.90963179	0.86128286
50000	0.89772474	0.85344705	0.8191937	0.79875631	0.88611251	0.83622804	0.86802014
55000	0.94403537	0.86590061	0.87543038	0.92978899	0.86886105	0.93413752	0.91926263
60000	0.92027395	0.86055153	0.79137672	0.81014059	0.86560875	0.85831237	0.86976274

Table 6. Recall Value for Movie Review Dataset

'Total number of samples'	Proposed 'Dragonfly+Hybrid'	DNN'	Naive Bayes'	Random Forest'	Decision Tree'	Elangovan et al.'	P Mudgil et al.'
10000	0.95038975	0.92053728	0.94582916	0.94332846	0.93284312	0.93242052	0.92628025
15000	0.92493812	0.90774367	0.80368881	0.84655245	0.89636093	0.91516065	0.87324677
20000	0.9551841	0.94609965	0.81414185	0.88162861	0.93599475	0.92640382	0.91503185
25000	0.94895446	0.89943624	0.92374442	0.8210611	0.8720917	0.92163887	0.88370802
30000	0.90246819	0.8318204	0.87418303	0.83905154	0.80154344	0.88584526	0.89157444
35000	0.8950017	0.8357004	0.86860028	0.83377584	0.85041815	0.86830881	0.83366873
40000	0.89566037	0.82394861	0.85023726	0.89427165	0.81458502	0.83231448	0.83983576
45000	0.89431135	0.81276814	0.87715554	0.77485697	0.88742534	0.82291392	0.88368025
50000	0.8982245	0.810134	0.79851722	0.76745219	0.84885315	0.84451592	0.88156917
55000	0.90439722	0.87333751	0.88379792	0.8645043	0.89198454	0.89424229	0.85527216
60000	0.93863265	0.84634187	0.91223913	0.86998053	0.90020426	0.89831233	0.93361444

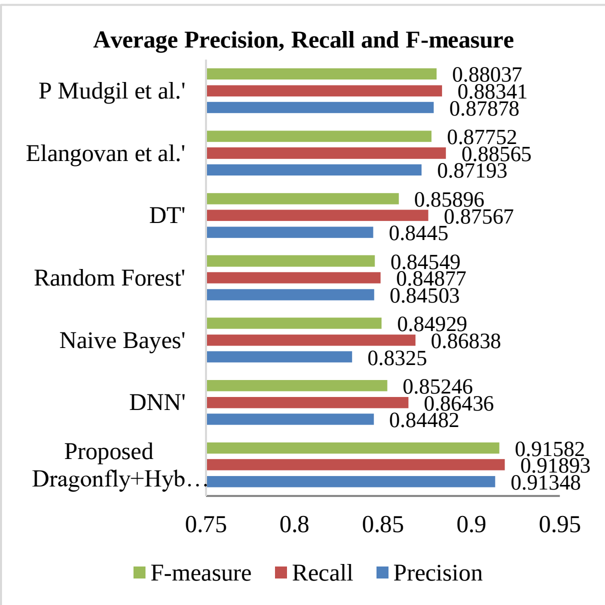
Table 7. F-measure for Movie Review Dataset

'Total number of samples'	Proposed 'Dragonfly+Hybrid'	DNN'	Naive Bayes'	Random Forest'	Decision Tree'	Elangovan et al.'	P Mudgil et al.'
10000	0.92369567	0.84714991	0.87472924	0.89402447	0.90167148	0.90824761	0.88492091
15000	0.93174924	0.891579	0.81105621	0.87680979	0.88807663	0.92299159	0.89818945
20000	0.92339061	0.86383982	0.79060532	0.83238074	0.88552292	0.86414049	0.89029901
25000	0.91947184	0.8340449	0.9037019	0.83407722	0.8119398	0.85803014	0.86883711
30000	0.89973359	0.85964045	0.87378074	0.82226661	0.84031158	0.86578631	0.88208793
35000	0.92292459	0.88698019	0.84594425	0.86963816	0.82993105	0.90582764	0.8698487
40000	0.89597628	0.79851397	0.81646192	0.84225399	0.81051992	0.83160118	0.8561268
45000	0.90588223	0.84100921	0.89004917	0.81112646	0.85057579	0.86410264	0.87233782
50000	0.89797455	0.83122667	0.80872333	0.78279141	0.86708275	0.84035155	0.87474219
55000	0.92379129	0.86960316	0.87959425	0.89595896	0.88027096	0.91375465	0.88611363
60000	0.92936265	0.85338756	0.84752065	0.83899492	0.88256761	0.87785693	0.9005582

Table 8. Accuracy for Movie Review Dataset

'Total number of samples'	Proposed 'Dragonfly+Hybrid'	DNN'	Naive Bayes'	Random Forest'	Decision Tree'	Elangovan et al.'	P Mudgil et al.'
10000	94.3900756	83.8660394	84.3029939	83.4269899	81.4384592	85.2985669	91.7701767
15000	94.7506074	83.4325011	81.8920603	84.4821828	86.6355318	85.9942879	92.0159684
20000	96.1006318	86.0230839	86.0125717	81.2432172	80.2162268	87.0687368	92.5373888
25000	94.9132278	83.1221865	84.6396564	80.8270864	84.9484219	87.9016578	93.7274283
30000	93.5174444	82.3509428	86.3360749	86.3849273	85.1538722	89.1804795	89.4882107
35000	92.6974059	80.0929824	80.6535936	80.7116858	82.0814277	85.3244589	92.41749
40000	96.4406838	80.7640795	82.0578114	80.3391314	83.6827173	90.4696394	91.8966888
45000	96.4612287	85.1887783	83.8467807	85.592722	80.5101971	91.601057	93.0590718
50000	95.4850255	85.2439202	81.6437884	86.068512	82.5843916	90.5255476	88.6066658
55000	93.7179014	84.4935843	85.4735238	81.6616111	82.8372079	90.4909952	94.7509587
60000	96.3066949	80.7682854	81.8632519	85.3827078	80.2606424	88.1621746	90.2106801

In the movie review dataset comprising a maximum of 60,000 records, various models were evaluated for their performance using key metrics such as Precision, Recall, F-measure, and Accuracy. The outcomes of the analysis is recorded in table 5, table 6, table 7 and table 8, respectively followed by graphical analysis in Fig. 4. The "Proposed Dragonfly + Hybrid" model standsout with impressive results, achieving an average Precision of 0.913, and a high average Recall of 0.918, signifying its effectiveness in capturing a significant portion of actual positive instances. F-measure of 0.915, highlighting its ability to provide a well-rounded trade-off between Precision and Recall. In terms of overall correctness, the "Proposed Dragonfly + Hybrid" model also excels with an average accuracy of 94.98%, indicating its proficiency in correctly classifying a substantial proportion of both positive and negative instances within the dataset. While other models like "DNN," "Naive Bayes," "Random Forest," and "DT" exhibit competitive performance across these metrics, the "Proposed Dragonfly+Hybrid" model consistently demonstrates robustness, making it a compelling choice for sentiment analysis tasks in the movie review dataset with up to 60,000 records. It's crucial to consider the specific requirements and trade-offs associated with each metric when selecting the most suitable model.



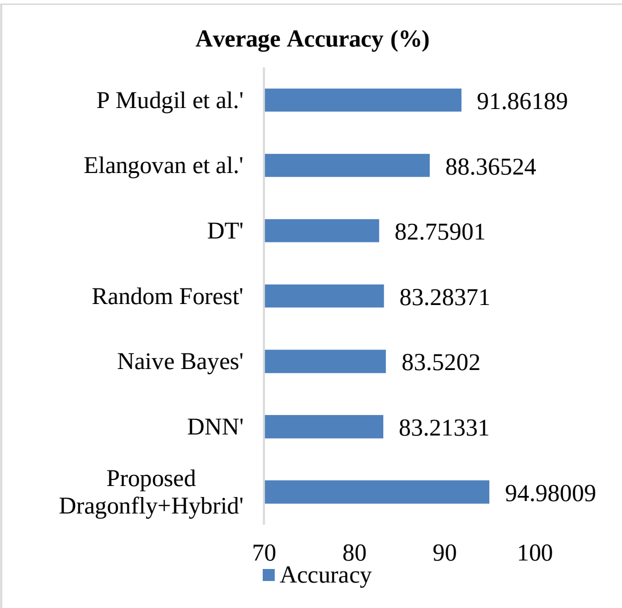


Fig. 4 Average Comparison Value for Movie Review Dataset

Table 9: Precision for Depression Dataset

'Total number of samples'	Proposed 'Dragonfly+Hybrid'	DNN'	Naive Bayes'	Random Forest'	Decision Tree'	Elangovan et al.'	P Mudgil et al.'
1000	0.97409055	0.95507365	0.91226852	0.89647366	0.9556072	0.96025581	0.94625768
2000	0.93757293	0.8594162	0.92669519	0.85912775	0.85002744	0.91340303	0.87984783
3000	0.94815288	0.89391685	0.8242609	0.9063182	0.86317607	0.90524019	0.9005094
4000	0.95005491	0.90870112	0.81368365	0.91578475	0.84450646	0.88530292	0.94163107
5000	0.93144989	0.85927768	0.79823527	0.9263414	0.91465767	0.88055087	0.91629856
6000	0.98465729	0.96050675	0.89937698	0.87359587	0.8713238	0.92143631	0.91233165
7000	0.97523515	0.88224725	0.96645528	0.9232931	0.87979116	0.96136855	0.95907645
8000	0.97890921	0.95294045	0.84314709	0.96606097	0.97229406	0.91424082	0.97700182
9000	0.97220932	0.88219063	0.93209552	0.93900164	0.96783556	0.89422925	0.89001306
10000	0.97486155	0.84340076	0.94301259	0.95922356	0.92302976	0.88268067	0.90758165

Table 10. Recall for Depression Dataset

'Total number of samples'	Proposed 'Dragonfly+Hybrid'	DNN'	Naive Bayes'	Random Forest'	Decision Tree'	Elangovan et al.'	P Mudgil et al.'
1000	0.96875031	0.95533036	0.8734873	0.9529377	0.94923435	0.95245096	0.94116165
2000	0.98250457	0.95532353	0.84783261	0.93631276	0.83690701	0.94351885	0.94692321
3000	0.96344704	0.94016043	0.91275381	0.88676205	0.91498883	0.92272737	0.89906075
4000	0.92874109	0.88450693	0.79838334	0.82232158	0.825137	0.88571618	0.85990122
5000	0.96661911	0.86473228	0.88819477	0.84929032	0.8904913	0.92806253	0.94440155
6000	0.96575899	0.8323253	0.90201374	0.93263305	0.86852606	0.92528903	0.9381687
7000	0.97965847	0.83992571	0.85229064	0.83315007	0.89075584	0.9771961	0.9156674
8000	0.97139608	0.83189622	0.93879062	0.87812507	0.8660229	0.93410723	0.89178891
9000	0.9769179	0.89258621	0.94048438	0.85903572	0.89842291	0.88028656	0.97531968
10000	0.95738963	0.8674997	0.93123263	0.86886314	0.89841982	0.89260837	0.92837792

Table 11. F-measure for Depression Dataset

'Total number of samples'	Proposed 'Dragonfly+Hybrid'	DNN'	Naive Bayes'	Random Forest'	Decision Tree'	Elangovan et al.'	P Mudgil et al.'
1000	0.97141309	0.95520199	0.89245681	0.92384373	0.95241012	0.95633746	0.94370279
2000	0.95951303	0.90483556	0.88551151	0.89606118	0.8434162	0.92821673	0.91215409
3000	0.95573878	0.91645566	0.8662532	0.89643348	0.88832758	0.91390013	0.8997845
4000	0.9392771	0.89644081	0.80596089	0.86654027	0.83470938	0.8855095	0.89891223
5000	0.94870868	0.86199635	0.84081566	0.8861441	0.90241272	0.90368264	0.93013783
6000	0.97511659	0.89183375	0.90069343	0.90214963	0.86992268	0.92335865	0.9250698
7000	0.9774418	0.86056646	0.90578984	0.87590845	0.88523955	0.96921772	0.93686937
8000	0.97513817	0.88831384	0.88840208	0.91999651	0.91608677	0.92406726	0.9324526
9000	0.97455792	0.88735798	0.93627116	0.89724048	0.93183838	0.88720313	0.93071572
10000	0.9660466	0.85528051	0.93708559	0.91181014	0.91055854	0.88761676	0.917862

Table 12. Accuracy for Depression Dataset

'Total number of samples'	Proposed 'Dragonfly+Hybrid'	DNN'	Naive Bayes'	Random Forest'	Decision Tree'	Elangovan et al.'	P Mudgil et al.'
1000	91.8773253	80.0547421	85.0604577	84.4223646	80.6901586	86.3737425	89.5229432
2000	95.9225626	81.4842159	84.9470065	81.1634249	81.1930242	85.5179634	94.8798464
3000	92.0805886	85.306787	84.79875	85.8269637	82.0945752	87.5172442	90.9703435
4000	91.1710381	84.5758309	83.2052043	85.3152917	80.7414429	86.4851424	89.2250682
5000	97.2607256	81.0751326	84.7581373	85.2403321	82.2808354	87.9056537	88.1437504
6000	97.528295	85.5626052	81.8032962	80.4493096	85.0064876	87.7353346	93.7017784
7000	94.5394563	83.1077491	84.4183211	81.5692005	82.710718	87.9477858	90.5925388
8000	94.9971811	81.8572543	82.6149467	81.2643182	82.4320635	89.3914255	93.6039106
9000	93.6831442	83.7103619	83.1614714	86.9027297	80.6095405	85.4686238	92.7868703
10000	97.1975162	80.9650829	80.292739	86.5776271	86.8904459	91.9181158	92.636673

The proposed work has also been evaluated for the depression dataset as mentioned in the methodology section. The performance analysis performed for precision, recall, f-measure and accuracy is summarized in table 9, table 10, table 11 and table 12, respectively, with a graphical analysis presented in Fig. 5. In the context of the depression dataset, which encompasses a comprehensive range of data with relevant mental health implications, the evaluation of different models based on key performance metrics provides valuable insights. The "Proposed Dragonfly + Hybrid" model consistently demonstrates impressive results across various metrics, showcasing its efficacy in addressing depression-related predictions. With an average Precision of 0.962, it excels in precisely identifying positive instances while minimizing the occurrence of false positives. Moreover, the model achieves an average Recall of 0.966, underlining its capacity to successfully identify a significant proportion of actual positive instances, a crucial aspect in mental health applications. These impressive Precision and Recall values translate into a high F-measure of 0.964, reflecting its ability to strike a balance between Precision and Recall. In terms of overall correctness, the "Proposed Dragonfly + Hybrid" model achieves an accuracy rate of 94.62%, emphasizing its ability to correctly classify a substantial proportion of both positive and negative instances within the depression dataset. This level of accuracy is essential in mental health-related applications, where precision is paramount. Comparatively, other models like "DNN," "Naive Bayes," "Random Forest," and "DT" display competitive performance across these metrics. However, the "Proposed Dragonfly + Hybrid" model consistently stands out, making it a convincing choice for tasks related to depression prediction and mental health analysis.

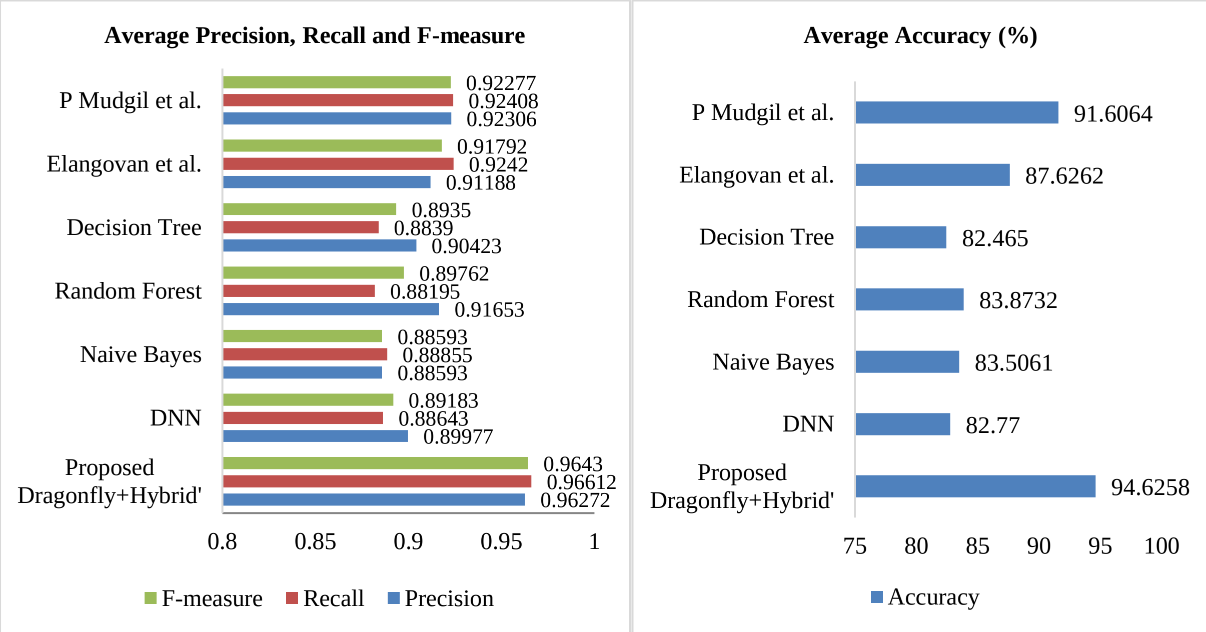


Fig 5. Average Comparison Value for Depression

5. Conclusion

The Dragonfly algorithm has been integrated into our Dragonfly + Hybrid Classifier, dramatically improving sentiment analysis accuracy across a variety of data sets. Dragonfly's effective feature set selection, resistance to noisy data, improved hyperparameters, and adaptability have all been critical in obtaining outstanding outcomes. The "Proposed Dragonfly + Hybrid" model consistently performs well in Precision, Recall, F-measure, and Accuracy.

It outperforms previous approaches for the Twitter dataset, including DNN, Naive Bayes, Random Forest, DT, Elangovan et al., and P Mudgil et al., with precision of 93.50%, recall of 92.96%, F-measure of 93.21%, and accuracy of 96.13%. In the movie review dataset, it achieves an average precision of 0.913, outperforming DNN (0.844), Naive Bayes (0.832), and Random Forest (0.845), with a high recall of 0.918 and an F-measure of 0.915, resulting in a 94.98% accuracy rate. In the depression dataset, the model obtains an average precision of 0.962 and recall of 0.966, exceeding DNN (0.886) and Naive Bayes (0.888), with an F-measure of 0.964 and 94.62% accuracy. These findings show that the "Proposed Dragonfly + Hybrid" model is versatile and effective across multiple data domains, making it a viable option for a variety of data-driven applications. While other models perform well, the "Proposed Dragonfly + Hybrid" model's steady and good performance demonstrates its versatility, making it an appealing option based on unique project requirements and metric priorities.

Acknowledgements

This research was not financially supported by any organization.

Author contributions

M.R. conceived of the presented idea. M.R. developed the theory and performed the computations, J.S verified the analytical methods. M.R. discussed the results and contributed to the final manuscript.

Conflicts of interest

The authors declare no conflicts of interest.

References

[1] S. L. Marie-Sainte and N. Alalyani, “Firefly algorithm based feature selection for Arabic text classification,” *Journal of King Saud University-Computer and Information Sciences*, vol. 32, no. 3, pp. 320–328, 2020.

[2] A. Ahad, B. Madhuravani, S. Kosuru, S. M. F. U. Haque, and M. A. Hussain, “An Intelligent Data Mining System for Tweet Opinion Analysis using Combined Cluster based Classification Approach,” *International Journal of Intelligent Systems and Applications in Engineering*, vol. 10, no. 4, pp. 604–609, Dec. 2022.

[3] A. H. Abdulhafiz, “Novel Opinion mining System for Movie Reviews,” *International Journal of Intelligent Systems and Applications in Engineering*, vol. 8, no. 2, pp. 94–101, Jun. 2020, doi: 10.18201/ijisae.2020261590.

[4] N. K. Laskari and S. K. Sanampudi, "Suggestion Key Phrase Extraction: A Fine-Grained Suggestion Mining from Opinion Reviews," *International Journal of Intelligent Systems and Applications in Engineering*, vol. 11, no. 4, pp. 164–171, Sep. 2023.

[5] P. Durga and D. Godavarthi, "A Review on Evaluation of Different Models for Classifying Sentiments from Twitter: Challenges and Applications," *International Journal of Intelligent Systems and Applications in Engineering*, vol. 12, no. 1, pp. 235–266.

[6] D. Elangovan and V. Subedha, "An Effective Feature Selection Based Classification model using Firefly with Levy and Multilayer Perceptron based Sentiment Analysis," in *2020 International Conference on Inventive Computation Technologies (ICICT)*, 2020, pp. 376–380.

[7] R. K. Patra, B. Patil, T. S. Kumar, G. Shivakanth, and B. M. Manjula, "Machine Learning Based Sentiment Analysis and Swarm Intelligence," in *2023 IEEE International Conference on Integrated Circuits and Communication Systems (ICICACS)*, 2023, pp. 1–8.

[8] S. S. Alotaibi, E. Alabdulkreem, S. Althahabi, M. A. Hamza, M. Rizwanullah, A. S. Zamani, A. Motwakel, and R. Marzouk, "Artificial fish swarm optimization with deep learning enabled opinion mining approach," *Computer Systems Science and Engineering*, vol. 45, no. 1, pp. 737–751, 2023.

[9] H. Chantar, M. Mafarja, H. Alsawalqah, A. A. Heidari, I. Aljarah, and H. Faris, "Feature selection using binary grey wolf optimizer with elite-based crossover for Arabic text classification," *Neural Computing and Applications*, vol. 32, pp. 12201–12220, 2020.

[10] D. Deepika and N. Balaji, "Effective heart disease prediction with Grey-wolf with Firefly algorithm-differential evolution (GF-DE) for feature selection and weighted ANN classification," *Computer Methods in Biomechanics and Biomedical Engineering*, vol. 25, no. 12, pp. 1409–1427, 2022.

[11] X. Yong and Y. L. Gao, "Improved firefly algorithm for feature selection with the ReliefF-based initialization and the weighted voting mechanism," *Neural Computing and Applications*, vol. 35, no. 1, pp. 275–301, 2023.

[12] T. Bezdan, M. Zivkovic, N. Bacanin, A. Chhabra, and M. Suresh, "Feature selection by hybrid brain storm optimization algorithm for covid-19 classification," *Journal of Computational Biology*, vol. 29, no. 6, pp. 515–529, 2022.

[13] P. Mudgil, P. Gupta, I. Mathur, and N. Joshi, "An improved context-aware analysis for sentimental Grass Hopper Optimization algorithm and its post affects on Twitter," *Journal of Intelligent \& Fuzzy Systems*, 2023, doi: 10.3233/JIFS-219430.

[14] R. Asgarnezhad, S. A. Monadjemi, and M. Soltanaghaei, "An application of MOGW optimization for feature selection in text classification," *The Journal of Supercomputing*, vol. 77, pp. 5806–5839, 2021.

[15] A. Alarifi, A. Tolba, Z. Al-Makhadmeh, and W. Said, "A big data approach to sentiment analysis using greedy feature selection with cat swarm optimization-based long short-term memory neural networks," *The Journal of Supercomputing*, vol. 76, pp. 4414–4429, 2020.

[16] M. Tubishat, M. A. Abushariah, N. Idris, and I. Aljarah, "Improved whale optimization algorithm for feature selection in Arabic sentiment analysis," *Applied Intelligence*, vol. 49, pp. 1688–1707, 2019.

[17] "Sentiment 140 dataset." available online at https://www.kaggle.com/datasets/kazanova/sentiment140 (accessed jan, 2024).

[18] "NLTK Movie Reviews Dataset." available onlne at https://www.kaggle.com/datasets/nltkdata/movie-review (accessed jan, 2024).

[19] "Depression Sentimental Analysis for Tweets Dataset." available at https://www.kaggle.com/datasets/gargmanas/sentimental-analysis-for-tweets (accessed jan, 2024).

[20] A. Madasu and S. Elango, "Efficient feature selection techniques for sentiment analysis," *Multimedia Tools and Applications 2019 79:9*, vol. 79, no. 9, pp. 6313–6335, Dec. 2019, doi: 10.1007/S11042-019-08409-Z.

[21] M. K. Vijaymeena and K. Kavitha, "A survey on similarity measures in text mining," *Machine Learning and Applications: An International Journal (MLAIJ)*, vol. 3, no. 1, 2016, doi: 10.5121/mlaij.2016.3103.

[22] A. Gupta, P. Tyagi, T. Choudhury, and M. Shamoon, "Sentiment Analysis Using Support Vector Machine," *Proceedings of the 4th International Conference on Contemporary Computing and Informatics, IC3I 2019*, pp. 49–53, Dec. 2019, doi: 10.1109/IC3I46837.2019.9055645.