I. J. Education and Management Engineering, **2026, 4, 16-28**
Published Online on August 8, 2026 by MECS Press (http://www.mecs-press.org/)
DOI: 10.5815/ijeme.2026.04.02

A Comparative Analysis of Evolutionary Metaheuristics: Genetic Algorithm vs. Particle Swarm Optimization for Feature Selection in High-Accuracy Fake News Detection

Nikita Garg*
Department of Computer Science & Engineering, HNB Garhwal University (A Central University), Srinagar Garhwal-246 174, Uttarakhand, India
E-mail: nikita.garg2706@gmail.com
ORCID iD: https://orcid.org/0009-0009-1418-2611
*Corresponding Author

Pritam Singh Negi
Department of Computer Science & Engineering, HNB Garhwal University (A Central University), Srinagar Garhwal-246 174, Uttarakhand, India
E-mail: negipritam@hnbgu.ac.in
ORCID iD: https://orcid.org/0009-0004-5703-6212

Received: March 25, 2026; Revised: April 15, 2026; Accepted: May 18, 2026; Published: August 8, 2026

Abstract: The proliferation of online misinformation necessitates highly accurate and computationally efficient automated systems for fake news detection. A primary impediment to system performance is the high dimensionality of textual features derived from techniques like TF-IDF, making optimal Feature Selection a critical step. This paper presents a detailed comparative experimental study of two prominent bio-inspired evolutionary metaheuristics, the Genetic Algorithm (GA) and Particle Swarm Optimisation (PSO) used as wrapper-based FS techniques for FND. The methodologies were rigorously tested across two distinct textual datasets: the complex, large-scale FakeNewsNet corpus and a moderate-scale general news dataset. The feature sets, once optimised, were evaluated using six standard Machine Learning (ML) classifiers. The GA-based FS approach, emphasising global exploration, achieved state-of-the-art accuracy of 99.91% with the Random Forest classifier on the FakeNewsNet dataset. In contrast, the PSO-based FS approach, valued for its rapid convergence, yielded a maximum accuracy of 93.29% with the Support Vector Machine (SVM) on the general news dataset. This analysis provides empirical evidence of the intrinsic trade-off between the algorithms: GA is superior for maximising accuracy in high-dimensional, complex textual spaces, while PSO offers a more efficient and practical solution for resource-constrained or moderate-scale FND tasks. The study demonstrates that evolutionary computation effectively enhances ML classifier performance by optimizing feature subsets.

Index Terms: Fake News Detection, Feature Selection, Genetic Algorithm, Particle Swarm Optimization, Evolutionary Computation, Wrapper Method, High Dimensionality, TF-IDF

1. Introduction

The rapid growth of digital media and online communication platforms has transformed the way information is created and shared. While social networking platforms like Facebook, X, and YouTube facilitate instant information access, they have also accelerated the dissemination of fake news and misinformation. Due to rapid content sharing and algorithm-driven recommendations, misleading information can spread quickly and influence public opinion, political decisions, and social stability [1,2]. Several recent incidents involving misinformation campaigns and false online reports have highlighted the need for reliable automated fake news detection systems [3]. Given the vast volume of daily online content, manual fact-checking is insufficient for real-time fake news identification. Consequently, machine learning and natural language processing techniques have become important research areas for automated fake news detection. Existing studies have applied classifiers such as Support Vector Machines, Naïve Bayes, Decision Trees,

Random Forests, and Deep Learning models for detecting deceptive textual content [4]. However, the performance of these models largely depends on the quality of selected textual features. Feature selection is an important step in fake news detection because textual datasets generally contain high-dimensional and redundant features. Irrelevant features increase computational complexity and may reduce classification performance. Traditional feature selection approaches, including filter-based and wrapper-based methods, have been widely used in text classification research [5]. However, these methods often face challenges related to scalability and computational efficiency. To overcome these limitations, researchers have increasingly adopted evolutionary and swarm intelligence optimization techniques for feature selection. Recent studies have shown that optimization-based approaches improve classification accuracy and reduce feature dimensionality in fake news detection systems [6–8]. Among these approaches, Genetic Algorithm (GA) and Particle Swarm Optimization (PSO) are widely used evolutionary metaheuristic techniques. Genetic Algorithm is inspired by the principles of natural evolution and uses operations such as selection, crossover, and mutation to generate optimal feature subsets. Previous studies have reported that GA-based feature selection improves classification accuracy in text mining applications [9]. In contrast, Particle Swarm Optimization is inspired by the collective behavior of bird flocks and updates particles based on local and global best solutions to explore the search space efficiently. PSO is known for faster convergence and lower computational complexity compared to several traditional optimization techniques [10].

Although both GA and PSO have demonstrated promising performance individually, limited studies provide a systematic comparative analysis of these optimization techniques under a unified framework. Many previous studies use different datasets, preprocessing methods, and evaluation strategies, making direct comparison difficult. Furthermore, several studies focus mainly on classification accuracy while ignoring computational efficiency and feature reduction capability. Therefore, the present study conducts a comparative analysis of Genetic Algorithm and Particle Swarm Optimization for feature selection in fake news detection. We extract textual features using TF-IDF and evaluate the optimized subsets generated by GA and PSO with multiple machine learning classifiers. The study aims to compare both optimization techniques in terms of classification accuracy, feature reduction performance, and computational efficiency to identify a suitable evolutionary approach for high-accuracy fake news detection systems.

2. Related Work

Fake news detection has attracted significant research attention due to the rapid spread of misinformation across online platforms. Existing studies have applied various machine learning and deep learning techniques for identifying deceptive news content. However, many approaches primarily focus on classification performance while giving limited attention to optimized feature selection.

Early studies relied mainly on feature extraction and classification models without incorporating feature optimization techniques. Akinyemi et al. utilized a Recurrent Neural Network (RNN) model for fake news detection and achieved an accuracy of 81% using extracted textual features [11]. Similarly, Kong et al. combined TF-IDF feature extraction with a neural network model and reported 90.3% accuracy [12]. Although these studies achieved satisfactory classification performance, the absence of feature selection may introduce redundant features and increase computational complexity.

Ensemble and tree-based learning approaches have also been widely explored for misinformation detection. Kaliyar et al. proposed a tree-based ensemble framework using content and context-level features, achieving 86% accuracy on the Fake News Challenge dataset [13]. Reddy et al. developed a hybrid ensemble model integrating multiple classifiers, including Naïve Bayes, Random Forest, Support Vector Machine, and K-Nearest Neighbors, and reported 94% accuracy across different datasets [14]. Despite improved robustness, these studies provided limited analysis of feature optimization and dimensionality reduction. Several recent studies have demonstrated strong fake news classification performance using machine learning and deep learning techniques. Alhariri et al. applied Random Forest and Decision Tree models to COVID-19 fake news datasets and achieved accuracies of 94.49% and 92.07%, respectively [15]. Ahmed et al. employed TF-IDF and n-gram-based feature extraction with Linear SVM, obtaining 92% classification accuracy [16]. However, these approaches relied on high-dimensional feature spaces without evaluating optimized feature subset selection, which may affect scalability and computational efficiency.

To improve feature selection performance, researchers have increasingly adopted evolutionary optimization techniques. Genetic Algorithm has shown promising performance in selecting informative textual features and reducing dimensionality in text classification tasks. Gupta et al. demonstrated improved tweet classification performance using GA-based feature optimization and achieved 91.65% accuracy [17]. Although GA improves feature subset quality and model generalization, it may experience slower convergence and increased computational cost when processing large-scale feature spaces.

Recent research has also explored deep learning and multimodal fake news detection frameworks. Alshariah et al. investigated fake image detection using CNN and AlexNet architectures and reported improved performance using AlexNet [18]. Giachanou et al. introduced a multimodal framework integrating textual, semantic, and image-based features using BERT and VGG-16 models [19]. Hamid et al. analyzed COVID-19 misinformation using BERT embeddings and Graph Neural Networks to capture semantic and structural relationships [20]. Although these approaches achieved high detection accuracy, they generally require large datasets and high computational resources.

Swarm intelligence techniques, particularly Particle Swarm Optimization, have emerged as efficient alternatives for feature selection in high-dimensional text classification problems. PSO optimizes feature subsets through collective learning mechanisms and efficient global search strategies. Recent studies have reported that PSO-based feature selection improves classification performance while reducing computational complexity in text mining applications [21]. However, limited studies have provided a systematic comparative evaluation of PSO and GA under identical fake news detection settings. Therefore, a significant research gap still exists in conducting a unified comparative analysis of Genetic Algorithm and Particle Swarm Optimization using consistent datasets, preprocessing methods, feature extraction techniques, and evaluation metrics. To address this limitation, the present study performs a comprehensive comparative analysis of GA and PSO for feature selection in fake news detection, focusing on classification accuracy, feature reduction capability, and computational efficiency.

3. Proposed Methodology

In this study, two evolutionary feature selection-based frameworks are proposed for fake news detection. The first framework employs GA, while the second utilises PSO. Both approaches aim to identify informative textual features to improve classification performance and computational efficiency in fake news detection systems. Different datasets were utilised to analyse the adaptability and optimization behaviour of both techniques under varying textual environments and feature distributions.

3.1. Proposed Model 1: Genetic Algorithm-Based Feature Selection

The GA-based framework utilizes the fakeNewsNet dataset for feature selection and classification. After pre-processing, textual features were extracted using TF-IDF vectorisation and optimized using GA. The selected feature subset was then evaluated using multiple ML classifiers. The workflow of the proposed framework is illustrated in Fig. 1.

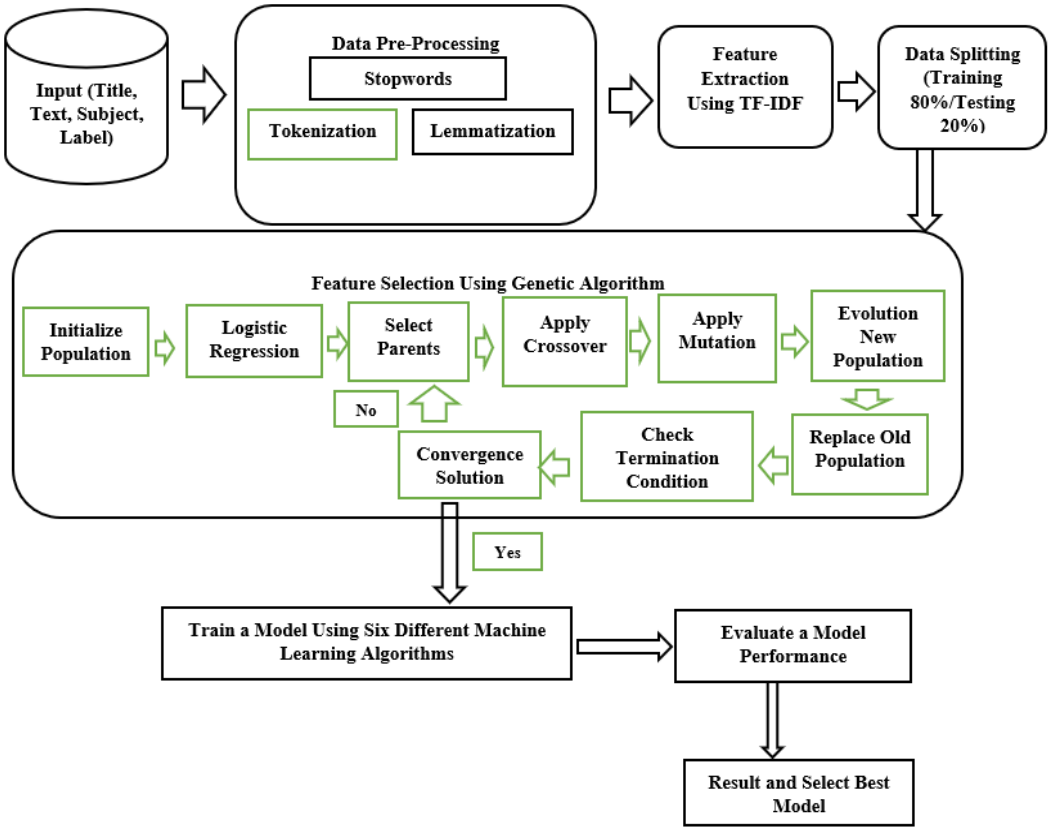

Fig. 1. Proposed Model 1

3.1.1. Dataset

For this study, the FakeNewsNet dataset obtained from Kaggle was utilized. The dataset contains fake and real news articles represented using title, text, subject, and label attributes. The combined dataset consisted of 44,898 records after merging the Fake and True news subsets. Labels were assigned as 1 for true and 0 for fake news for binary

classification. The datasets were divided into training and testing sets using an 80:20 ratio to ensure reliable model evaluation.

3.1.2. Data Pre-Processing

Text pre-processing was performed using the NLTK library to improve data quality and consistency. The preprocessing steps included tokenization, stopwords removal, lowercasing, punctuation removal, URL removal, and lemmatization. These operations reduced redundancy and prepared the textual data for feature extraction and classification.

3.1.3. Feature Extraction

Feature Extraction was performed using TF-IDF vectorization to convert textual data into numerical representation suitable for ML models. A total of 1,000 features were extracted to maintain a balance between computational efficiency and feature representation. The resulting TF-IDF matrix dimensions were (4489,1000).

TF-IDF Formula

$$TF\text{-}IDF(t,d) = TF(t,d) \times IDF(t) \quad (1)$$

Where:

- TF(t,d) represents the term frequency of term *t* in document *d*.
- IDF(t) represents the inverse document frequency.

3.1.4. Data Splitting

The dataset was divided into training and testing sets using the train_test_split function with an 80:20 ratio and random_state=42 to ensure reproducibility and reliable model evaluation.

3.1.5. Feature Selection

Feature selection was performed using GA to identify informative features and reduce dimensionality. GA was selected due to its strong global search capability and ability to avoid local optima in high-dimensional search spaces. The GA initialized a population of 50 individuals, where each individual represented a binary feature subset. Fitness evaluation was performed using logistic regression classifier accuracy. Tournament selection, two-point crossover, and mutation operations were applied iteratively for 10 generations. The optimal feature subset selected by GA contained 494 features.

Pseudocode for Genetic Algorithm-Based Feature Selection

Input:
  TF-IDF Feature Matrix X
  Target Labels Y
  Population Size = 50
  Generations = 10
  Mutation Rate = 0.10

Output:
  Optimal Feature Subset

Begin
1. Initialize random population
2. Evaluate fitness using Logistic Regression
3. Apply tournament selection
4. Perform crossover operation
5. Apply mutation operation
6. Generate new population
7. Repeat for 10 generations
8. Select best feature subset
End

The pseudocode representation of the Genetic Algorithm is provided for methodological clarity and reproducibility.

3.1.6. Classification Algorithm

The selected features were evaluated using six machine learning classifiers, including Decision Tree (DT), Random Forest (RF), Logistic Regression (LR), Support Vector Machine (SVM), K-Nearest Neighbors (KNN), and Multilayer Perceptron (MLP).

3.1.6.1. Decision Tree (DT):

A decision tree is a widely used machine learning model for both classification and regression tasks. It operates by splitting the data into smaller subsets based on the most informative features, creating a tree-like structure. The tree consists of a root node, decision nodes, and leaf nodes. The root node represents the entire dataset, while decision nodes test specific attributes, and leaf nodes provide the final output or prediction. The tree is built through recursive feature selection, often using metrics such as information gain for classification tasks or variance reduction for regression. To avoid overfitting, pruning is employed, removing unnecessary branches. Although decision trees are straightforward and capable of capturing complex, non-linear relationships, they can suffer from overfitting if the tree becomes too deep, which can be addressed by techniques like limiting tree depth or pruning.

3.1.6.2. Random Forest (RF):

Random Forest is an ensemble learning method used for classification and regression. It builds multiple decision trees using random subsets of data and features. The final prediction is made by aggregating the outputs of all the trees, either by majority vote for classification or averaging for regression. This approach improves accuracy and reduces overfitting. Random Forest handles complex, high-dimensional data well and is robust to noise, though it can be computationally expensive and harder to interpret due to the large number of trees involved.

3.1.6.3. Logistic Regression (LR):

Logistic Regression is a binary classification technique used to predict one of two possible outcomes (e.g., 0/1 or true/false). Unlike linear regression, it estimates the probability of an event using the sigmoid function, which converts the weighted sum of input features into a value between 0 and 1. If the probability is greater than 0.5, the instance is classified as 1, and below, it is classified as 0. The model adjusts its coefficients during training to minimize the difference between predicted probabilities and actual outcomes using a logarithmic loss function (cross-entropy).

3.1.6.4. Support Vector Machine (SVM):

Support Vector Machine (SVM) is a supervised learning algorithm used for classification and regression tasks. Its goal is to identify the best hyperplane that separates different classes of data with the maximum margin. The support vectors are the data points that are closest to this hyperplane and determine its position. In cases where data is not linearly separable, SVM utilizes the kernel trick to transform the data into a higher-dimensional space, making it easier to find a separating hyperplane. Popular kernel functions include linear, polynomial, and radial basis function (RBF). The regularization parameter C controls the balance between maximizing the margin and minimizing misclassification. When the data is linearly separable, SVM optimizes the hyperplane's position by minimizing the weight vector's norm, while ensuring all data points are correctly classified.

3.1.6.5. K-Nearest Neighbors (KNN):

K-Nearest Neighbors (KNN) is a simple machine learning algorithm used for classification and regression. It works by finding the k nearest data points to a new point and using their class (for classification) or average value (for regression) to make a prediction. The algorithm relies on distance metrics, such as Euclidean or Manhattan distance, to measure proximity. The choice of k influences the model's sensitivity to noise and its ability to capture details. Although easy to implement, KNN can be computationally expensive and memory-intensive for large datasets.

3.1.6.6. MultiLayer Perception (MLP):

A Multilayer Perceptron (MLP) is a neural network with multiple layers, including an input layer, hidden layers, and an output layer. Data passes through these layers, where each neuron processes information using weights, biases, and activation functions like ReLU or Sigmoid. MLP is trained with backpropagation, adjusting weights based on the error between predictions and actual outcomes. While effective for complex tasks, MLP can be computationally intensive and prone to overfitting without regularization. Despite these challenges, it is a powerful model for learning from data.

3.1.7. Performance Evaluation

The performance of the GA-Based feature selection framework was assessed using widely accepted classification evaluation metrics, namely accuracy, precision, recall, and F1-score. These metrics help determine how effectively the selected feature subset improves the predictive performance of the classifier.

3.1.7.1. Accuracy:

Accuracy represents the proportion of correctly classified instances among all predictions made by the GA-Based model. It provides an overall indication of classification performance but should be interpreted carefully when dealing with imbalanced datasets [20].

3.1.7.2. Precision:

Precision measures the proportion of true positive predictions among all positive predictions. A high precision value indicates that the classifier generates fewer false positive predictions [20].

3.1.7.3. Recall:

Recall evaluates the ability of the classifier to identify all actual positive samples. It reflects how effectively the GA-Based model minimizes false negatives and captures relevant instances [20].

3.1.7.4. F1-Score:

The F1-Score combines precision and recall in a single metric by calculating their harmonic mean. It provides a balanced assessment of the classifier, particularly when both false positives and false negatives are important [20].

3.2. Proposed Model 2: Particle Swarm Optimization-Based Feature Selection

A Particle Swarm Optimization (PSO)-based framework was proposed for fake news detection. The preprocessing, feature extraction, and data splitting procedures for the PSO framework mirrored those of the GA framework to ensure methodological consistency.

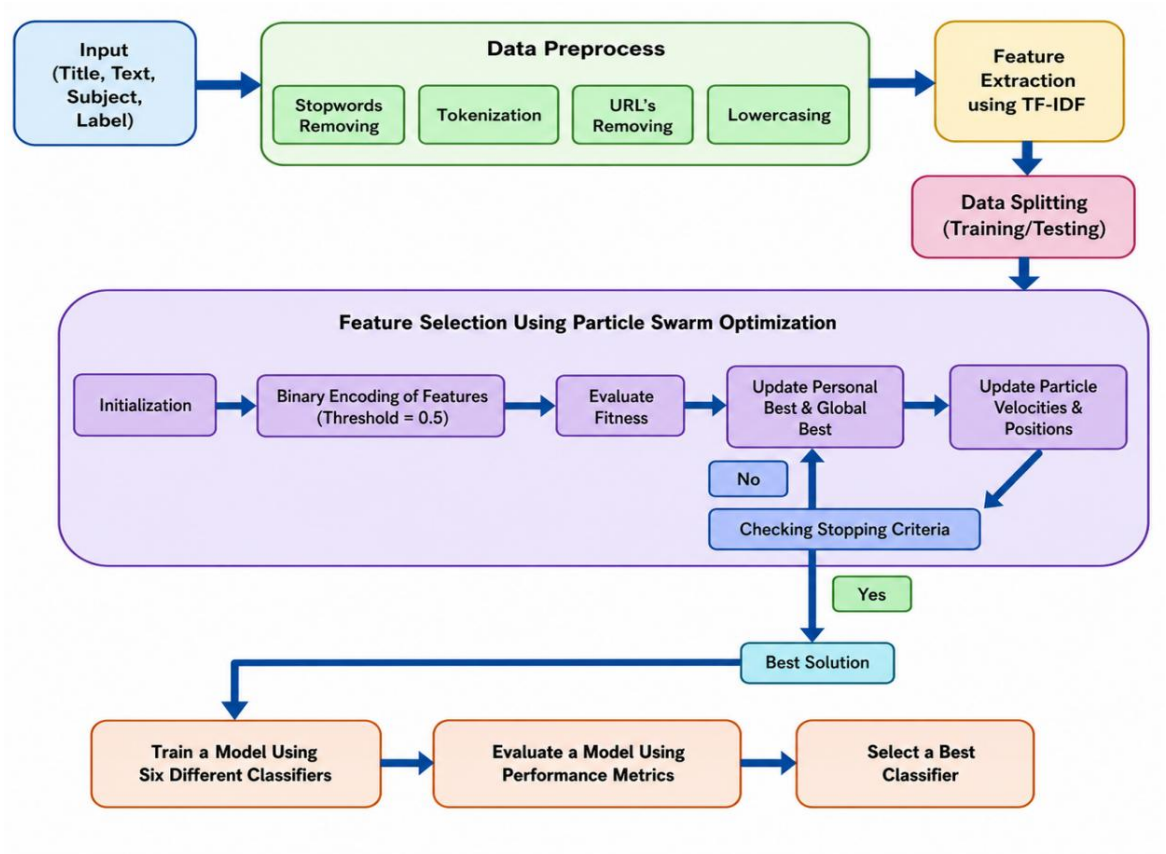

Fig. 2. Proposed Model 2

3.2.1. Dataset Collection

For the PSO framework, a Kaggle-based fake news dataset containing 6,335 records was utilised. The dataset included title, text, and label attributes representing fake and real news articles.

3.2.2. Data Pre-processing

The title and text columns were merged into a unified textual representation. Preprocessing operations included stopword removal, punctuation removal, URL removal, tokenization, and lowercasing to improve textual consistency and reduce noise.

3.2.3. Feature Extraction

TF-IDF vectorization was applied to convert textual data into numerical feature representations. A total of 5,000 features were extracted for optimization and classification.

TF-IDF Formula

$$TF\text{-}IDF(t,d) = TF(t,d) \times IDF(t) \tag{2}$$

Where:

- TF(t,d) represents the term frequency of term *t* in document *d*.
- IDF(t) represents the inverse document frequency.

3.2.4. Data Splitting

The dataset was divided into training and testing sets using an 80:20 ratio with random_state=42 to ensure reproducibility and reliable evaluation.

3.2.5. Feature Selection Using Particle Swarm Optimization

Feature selection was performed using Particle Swarm Optimization (PSO), where each particle represented a candidate feature subset. The optimization process iteratively updated particle positions based on local and global best solutions to identify informative features while reducing dimensionality. The final optimized subset contained 2,500 selected features.

The fitness function evaluated feature subsets using Logistic Regression classification accuracy. PSO was executed iteratively to balance exploration and exploitation for identifying optimal feature subsets.

```
def pso_fitness_function(weights):
    selected_features = (weights > 0.5).astype(int)
    if not selected_features.any(): return 1e5
    model = LogisticRegression().fit(x_train[:, selected_features], y_train)
   return -accuracy_score(y_test, model.predict(x_test[:, selected_features]))
pso_weights, _ = pso(pso_fitness_function, [0]*x_train.shape[1], [1]*x_train.shape, swarmsize==20, maxiter=10).
```

3.2.6. Classification Algorithms

The optimized feature subset was evaluated using six machine learning classifiers, including K-Nearest Neighbors (KNN), Naïve Bayes (NB), Gradient Boosting (GB), Random Forest (RF), Support Vector Machine (SVM), and Logistic Regression (LR).

3.2.6.1. K-Nearest Neighbors (KNN):

KNN is an instance-based learning algorithm used for classification and regression. It classifies data points by looking at the 'K' nearest neighbors and selecting the most common class. Euclidean distance is typically used to calculate the distance between points. KNN doesn't require a training phase but can be computationally expensive during prediction, as it calculates distances to all training points. It is also sensitive to noisy data and irrelevant features, which can affect its performance.

3.2.6.2. Naïve Bayes (NB):

Naive Bayes is a probabilistic classification algorithm based on Bayes' Theorem, assuming that features are independent given the class. This assumption simplifies the calculation of probabilities. While the assumption may not always hold, Naive Bayes performs well, especially in tasks like text classification. It calculates the probability of each class using feature likelihoods and class priors. The algorithm is efficient and performs well with large datasets, though it may struggle when features are not truly independent.

3.2.6.3. Gradient Boosting (GR):

Gradient Boosting is an ensemble learning approach where decision trees are built in sequence, and each subsequent tree focuses on minimising the errors made by the previous trees. The final prediction is a weighted sum of all the tree predictions, and gradient descent is used to minimize the loss function. While it is a powerful and accurate technique, it can be computationally intensive and may overfit if not carefully tuned.

3.2.6.4. Random Forest (RF):

Random Forest is an ensemble method that builds multiple decision trees, each trained on a random subset of data and features. Predictions are made by combining the results from all the trees, using majority voting for classification and averaging for regression. This method reduces overfitting, making it more reliable than a single decision tree. It is effective across various data types but can be computationally demanding and less interpretable than individual trees.

3.2.6.5. Support Vector Machine (SVM):

Support Vector Machine is a powerful classification algorithm that finds the optimal hyperplane to separate data points of different classes in a multi-dimensional space. Its goal is to maximize the margin between the data points and the hyperplane. SVM can address both linear and non-linear problems by employing kernel techniques, such as polynomial or radial basis function (RBF) kernels, which transform the data into a higher-dimensional space. While SVM is effective for complex problems, it can be computationally intensive, particularly with large datasets. Additionally, its performance is sensitive to the choice of kernel and hyperparameters.

3.2.6.6. Logistic Regression (LR):

Logistic Regression is a model used for binary classification that estimates the probability of a data point belonging to a specific class. It uses a logistic function to transform values between 0 and 1 and fits a linear equation to the input features. While efficient and easy to interpret, it assumes a linear relationship between features and the target, which may not be ideal for complex datasets.

3.2.7. Performance Evaluation

The performance of the PSO-based framework was evaluated using Accuracy, Precision, Recall, F1-Score, and Confusion Matrix metrics. The evaluation was performed using accuracy, precision, recall and F1-score. These metrics examine how effectively the optimized feature subset enhances the classifier's prediction capability.

3.2.7.1 Accuracy:

Accuracy measures the percentage of correctly predicted samples after applying PSO-Based features optimization. It reflects the overall effectiveness of the classification model across the entire dataset [20].

3.2.7.2 Precision:

Precision indicates the proportion of correctly identified positive samples among all samples predicted as positive. In the PSO-Based framework, it evaluates the model's ability to reduce false positive classification [20].

3.2.7.3 Recall:

Recall determines the proportion of actual positive instances correctly recognised by the classifier. A higher recall demonstrates that the PSO-Optimised model successfully identifies the most relevant positive samples.[20].

3.2.7.4 F1-Score:

The F1-score provides a comprehensive evaluation by balancing precision and recall. It is particularly useful when the dataset contains uneven class distributions or when both types of classification errors need equal consideration. [20].

3.3. Dataset Difference Justification

Different datasets were utilized for the GA and PSO frameworks to evaluate the optimization behavior and adaptability of both techniques under varying dataset characteristics. Although this may limit direct comparability, the study primarily focuses on analyzing the effectiveness of evolutionary feature selection techniques in distinct textual environments.

3.4. Limitation Section

One limitation of the present study is that the GA and PSO frameworks were evaluated using different datasets. Future work may focus on evaluating both optimization techniques using identical datasets and unified experimental settings for more consistent comparative analysis.

4. Experimental Results and Analysis

Two evolutionary feature selection-based frameworks were experimentally evaluated for fake news detection using Genetic Algorithm (GA) and Particle Swarm Optimization (PSO). Both frameworks employed TF-IDF vectorization for feature extraction and multiple machine learning classifiers for performance evaluation.

4.1. GA-Based Feature Selection Model

The GA-based framework utilized the FakeNewsNet dataset containing fake and true news articles. The title, text, and subject columns were selected for analysis. Text preprocessing included lowercasing, stopword removal, punctuation removal, URL removal, and lemmatization. TF-IDF vectorization was applied to convert textual data into numerical feature representations, resulting in a feature matrix of dimensions (8980 × 5000). Genetic Algorithm was then employed to optimize the feature subset and reduce dimensionality while preserving informative features. The optimized feature subset was evaluated using six machine learning classifiers, including Decision Tree (DT), Random Forest (RF), Logistic Regression (LR), Support Vector Machine (SVM), K-Nearest Neighbors (KNN), and Multilayer Perceptron (MLP. Performance evaluation was conducted using Accuracy, Precision, Recall, F1-Score, and Support metrics. The experimental results demonstrated that GA-based feature selection improved classification performance by eliminating redundant and irrelevant features. The comparative performance of classifiers is summarized in the corresponding result table 1.

Table 1. Performance of Six Classifiers (GA-Based Feature Selection)

Classifiers	Accuracy	F1-Score	Recall	Support
Decision Tree	0.99588	0.995874	0.995871	8980.0
Random Forest	0.99910	0.99108	0.99116	8980.0
Logistic Regression	0.992539	0.992530	0.992581	8980.0
Support Vector Machine	0.996659	0.996655	0.996695	8980.0
K-Nearest Neighbors	0.877728	0.877298	0.876730	8980.0
Multilayer Neural Network	0.998664	0.998662	0.998670	8980.0

4.2. PSO-Based Feature Selection Model

The PSO-based framework utilized a Kaggle news dataset containing title, text, and label attributes. Preprocessing operations included stopword removal, text cleaning, lowercasing, punctuation removal, and merging of textual columns. TF-IDF vectorization was applied for feature extraction, followed by Particle Swarm Optimization for feature selection. PSO optimized the feature space by selecting informative features while reducing dimensionality. The optimized feature subset was evaluated using six machine learning classifiers, including K-Nearest Neighbors (KNN), Naïve Bayes (NB), Gradient Boosting (GB), Random Forest (RF), Support Vector Machine (SVM), and Logistic Regression (LR). Experimental evaluation was performed using Accuracy, Precision, Recall, F1-Score, and Confusion Matrix metrics. Among the evaluated classifiers, SVM achieved the highest performance within the PSO-based framework. The detailed classifier results are presented in the corresponding performance table 2.

Table 2. Performance of Six Classifiers (PSO-Based Feature Selection)

Classifier	Accuracy	Precision	Recall	F1-Score
K-Nearest Neighbor (KNN)	0.8098	0.8116	0.8098	0.8096
Naïve Bayes (NB)	0.8824	0.8824	0.8823	0.8824
Gradient Boosting (GB)	0.8832	0.8832	0.8334	0.8832
Random Forest (RF)	0.9100	0.9123	0.9124	0.9124
Support Vector Machine (SVM)	0.9329	0.9327	0.927	0.929
Logistic Regression (LR)	0.9187	0.9186	0.9186	0.9186

4.3. Comparative Analysis

Both Genetic Algorithm and Particle Swarm Optimization demonstrated effective feature selection capability for fake news detection. GA provided strong optimization performance through evolutionary operations such as selection,

crossover, and mutation, resulting in improved classification accuracy for several classifiers. In contrast, PSO achieved efficient feature optimization through collective particle learning and faster convergence behavior. The experimental findings indicate that GA-based models achieved strong predictive performance with complex classifiers such as MLP, whereas PSO-based models demonstrated effective performance with SVM classifiers. Overall, both optimization techniques significantly improved feature selection efficiency and classification performance.

One limitation of the present study is that different datasets were utilized for the GA and PSO frameworks. Although these limits direct experimental comparability, the objective of the study was to analyze the adaptability and optimization effectiveness of evolutionary feature selection techniques under different textual environments.

5. Conclusion

This study proposed two feature selection-based fake news detection frameworks using Genetic Algorithm and Particle Swarm Optimization. Both frameworks utilized TF-IDF feature extraction and machine learning classifiers to improve fake news classification performance. The GA-based framework demonstrated strong classification performance by selecting informative features and reducing redundant information from high-dimensional textual data. Similarly, the PSO-based framework effectively optimized feature subsets while improving computational efficiency and classification accuracy. Experimental results demonstrated that both evolutionary optimization techniques significantly improved fake news detection performance. Among the evaluated classifiers, the GA-based framework achieved the highest overall classification accuracy, while the PSO-based framework demonstrated efficient feature optimization with strong classifier performance. The comparative analysis confirms that evolutionary and swarm intelligence-based feature selection approaches can effectively enhance machine learning performance for fake news detection tasks.

6. Limitations and Future Scope

Despite the strong experimental performance of the proposed frameworks, certain limitations remain. One major limitation is that the GA-based and PSO-based frameworks were evaluated . Additionally, both frameworks primarily relied on textual features and did not incorporate multimodal information such as images, videos, or social network interactions. The GA-based framework required higher computational effort due to iterative evolutionary operations, whereas PSO occasionally risked premature convergence during optimization. Future research may focus on developing hybrid GA-PSO optimization frameworks to combine the strengths of both techniques for improved feature selection and convergence performance. Furthermore, integrating deep learning embeddings, multimodal datasets, multilingual news sources, and real-time fake news streams may improve scalability and practical applicability in real-world environments.

All the Declarations and Statements

Author Contributions Statement

Nikita Garg – Carried out the research work, including conceptualization, methodology development, data analysis, and manuscript writing.
Pritam Singh Negi – Supervised the research work, provided continuous guidance, and contributed to reviewing and refining the manuscript.

All authors have read and agreed to the published version of the manuscript.

Conflict of Interest Statement

The authors declare no conflicts of interest.

Funding Declaration

The present manuscript has no funding source to declare.

Data Availability Statement

This study analyzed publicly available datasets. The results obtained and datasets can be found here: “kaggle”, accessed on “august”.

Ethical Declarations

It does not involve direct interaction with human participants or animals. Therefore, it does not require formal ethics approval or consent to participate.

Acknowledgments

The authors would like to express their sincere appreciation to HNB Garhwal University, Srinagar Garhwal (Uttarakhand), India, for providing the necessary resources and institutional support for this research. The authors are also grateful to Dr. Pritam Singh Negi for his valuable guidance, support, and insightful suggestions throughout the course of this study, which significantly contributed to the development of this work.

Declaration of Generative AI in Scholarly Writing

N/A

Abbreviations

The following abbreviations are used in this manuscript:

AI - Artificial Intelligence
NLP - Natural Language Processing
DL - Deep Learning
ML: Machine Learning.
FN: Fake News.
FND: Fake News Detection.
SVM: Support Vector Machine.
RF: Random Forest.
GB: Gradient Boosting
LR**:** Logistic Regression.
GA: Genetic Algorithm
PSO: Particle Swarm Intelligence

Appendix A/B/C…, with appendix title

APPENDIX A GENETIC ALGORITHM PSEUDOCODE

Step 1: Initialize population randomly
Step 2: Evaluate fitness of each individual
Step 3: Select best individuals
Step 4: Apply crossover and mutation
Step 5: Generate new population
Step 6: Repeat until stopping criteria met
Step 7: Return best feature subset

APPENDIX B PARTICLE SWARM OPTIMIZATION PSEUDOCODE

Step 1: Initialize particles randomly
Step 2: Evaluate particle fitness
Step 3: Update personal best and global best
Step 4: Update velocity and position
Step 5: Repeat until stopping criteria met
Step 6: Return optimal feature subset

References

[1] World Economic Forum. (2024). The global risks report 2024. World Economic Forum. https://doi.org/10.5822/978-1-944835-48-8.
[2] UNESCO. (2023). Guidelines for the governance of digital platforms: Safeguarding freedom of expression and access to information. UNESCO Publishing. https://doi.org/10.54675/XMNM1693.
[3] World Health Organization. (2023). Managing misinformation in the digital age. World Health Organization. https://doi.org/10.2471/BLT.23.290258.
[4] Shu, K., Sliva, A., Wang, S., Tang, J., & Liu, H. (2017). Fake news detection on social media: A data mining perspective. ACM SIGKDD Explorations Newsletter, 19(1), 22–36. https://doi.org/10.1145/3137597.3137600.
[5] Guyon, I., & Elisseeff, A. (2003). An introduction to variable and feature selection. Journal of Machine Learning Research, 3, 1157–1182. https://doi.org/10.1162/153244303322753616.
[6] Islam, M. R., Liu, S., Wang, X., & Xu, G. (2020). Deep learning for misinformation detection on online social networks: A survey and new perspectives. Social Network Analysis and Mining, 10(1), 82. https://doi.org/10.1007/s13278-020-00696-x.

[7] Sahoo, S. R., & Gupta, B. B. (2021). Multiple features based approach for automatic fake news detection on social networks using deep learning. Applied Soft Computing, 100, 106983. https://doi.org/10.1016/j.asoc.2020.106983.
[8] Kaliyar, R. K., Goswami, A., & Narang, P. (2022). Multiclass fake news detection using ensemble learning and content-based features. Applied Soft Computing, 115, 108243. https://doi.org/10.1016/j.asoc.2021.108243.
[9] Ahmed, H., Traore, I., & Saad, S. (2018). Detecting opinion spams and fake news using text classification. Security and Privacy, 1(1), e9. https://doi.org/10.1002/spy2.9.
[10] Ahmed, B., Ali, G., Hussain, A., Baseer, A., & Ahmed, J. (2021). Analysis of text feature extractors using deep learning on fake news. Engineering, Technology & Applied Science Research, 11(2), 7001–7005. https://doi.org/10.48084/etasr.4069.
[11] Giachanou, A., & Crestani, F. (2020). Like it or not: A survey of Twitter sentiment analysis methods. ACM Computing Surveys, 49(2), 1–41. https://doi.org/10.1145/2938640.
[12] Zhou, X., & Zafarani, R. (2020). A survey of fake news: Fundamental theories, detection methods, and opportunities. ACM Computing Surveys, 53(5), 1–40. https://doi.org/10.1145/3395046.
[13] Oshikawa, R., Qian, J., & Wang, W. Y. (2020). A survey on natural language processing for fake news detection. Proceedings of the 12th Language Resources and Evaluation Conference, 6086–6093. https://doi.org/10.48550/arXiv.1811.00770.
[14] Bondielli, A., & Marcelloni, F. (2019). A survey on fake news and rumour detection techniques. Information Sciences, 497, 38–55. https://doi.org/10.1016/j.ins.2019.05.035.
[15] Cui, L., Lee, D., & Zhang, Y. (2019). Detecting fake news on social media with graph neural networks. arXiv Preprint. https://doi.org/10.48550/arXiv.1902.06673.
[16] Khan, J. Y., Khondaker, M. T. I., Afroz, S., Uddin, G., & Iqbal, A. (2020). A benchmark study of machine learning models for online fake news detection. Machine Learning with Applications, 1, 100032. https://doi.org/10.1016/j.mlwa.2020.100032.
[17] Ruchansky, N., Seo, S., & Liu, Y. (2017). CSI: A hybrid deep model for fake news detection. Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, 797–806. https://doi.org/10.1145/3132847.3132877.
[18] Wang, W. Y. (2017). "Liar, liar pants on fire": A new benchmark dataset for fake news detection. Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, 422–426. https://doi.org/10.18653/v1/P17-2067.
[19] Shu, K., Mahudeswaran, D., Wang, S., Lee, D., & Liu, H. (2018). FakeNewsNet: A data repository with news content, social context, and dynamic information for studying fake news on social media. Big Data, 8(3), 171–188. https://doi.org/10.1089/big.2020.0062.
[20] Zellers, R., Holtzman, A., Rashkin, H., Bisk, Y., Farhadi, A., Roesner, F., & Choi, Y. (2019). Defending against neural fake news. Advances in Neural Information Processing Systems, 32. https://doi.org/10.48550/arXiv.1905.12616.
[21] Roy, A., Basak, K., Ekbal, A., & Bhattacharyya, P. (2021). A deep ensemble framework for fake news detection and classification. Applied Soft Computing, 106, 107414. https://doi.org/10.1016/j.asoc.2021.107414.

Authors' Profiles

Nikita Garg is a Research Scholar in the Department of Computer Science at Hemvati Nandan Bahuguna Garhwal University, a Central University, India. She is currently pursuing her Ph.D. in Computer Science, expected to be completed in September 2026. Her major field of study includes Natural Language Processing, Machine Learning, and Fake News Detection. I have been actively engaged in research work focusing on feature selection techniques using evolutionary algorithms and swarm intelligence for detecting fake news. Her research contributions include theoretical framework development in the area of information credibility analysis and text classification. Her current research interests include Natural Language Processing, Machine Learning, Swarm Intelligence, and Evolutionary Computation. I have been honoured with a Young Scientist Award for my outstanding academic and research contributions. I participated in various academic and research activities and continue to contribute to advanced studies in computational intelligence and text analytics.

Dr. Pritam Singh Negi is an Assistant Professor in the Department of Computer Science and Engineering at Hemvati Nandan Bahuguna Garhwal University, Chauras Campus, India. He holds an MCA degree and a Ph.D. in Information Technology and has also qualified UGC-NET and USET examinations. His major field of study includes Natural Language Processing, Network Security, and Computer Networks. He has more than 17 years of teaching experience and 8 years of research experience in the field of computer science. He has contributed significantly to research areas such as natural language processing, information retrieval, machine learning, cloud computing, and intrusion detection systems. He has published several research papers in reputed international journals covering topics including sentence boundary disambiguation, text summarization, pattern recognition, and cybersecurity applications. His current research interests include Natural Language Processing, Machine Learning, Network Security, and Intelligent Computing Systems. Dr. Negi is a member of the International Association of Computer Science and Information Technology (IACSIT). He has actively contributed to academic research, publications, and mentoring of research scholars in advanced computing domains.

How to cite this paper

Nikita Garg, Pritam Singh Negi, "A Comparative Analysis of Evolutionary Metaheuristics: Genetic Algorithm vs. Particle Swarm Optimization for Feature Selection in High-Accuracy Fake News Detection", International Journal of Education and Management Engineering (IJEME), Vol.16, No.4, pp. 16-28, 2026. DOI:10.5815/ijeme.2026.04.02.