



**ARTICLE**

# Ensemble Filter-Wrapper Text Feature Selection Methods for Text Classification

Oluwaseun Peter Ige<sup>1,2</sup> and Keng Hoon Gan<sup>1,\*</sup>

<sup>1</sup>School of Computer Sciences, Universiti Sains Malaysia, Gelugor, 11800, Malaysia

<sup>2</sup>Universal Basic Education Commission, Abuja, 900284, Nigeria

\*Corresponding Author: Keng Hoon Gan. Email: khgan@usm.my

Received: 30 April 2024 Accepted: 31 July 2024 Published: 27 September 2024

## ABSTRACT

Feature selection is a crucial technique in text classification for improving the efficiency and effectiveness of classifiers or machine learning techniques by reducing the dataset's dimensionality. This involves eliminating irrelevant, redundant, and noisy features to streamline the classification process. Various methods, from single feature selection techniques to ensemble filter-wrapper methods, have been used in the literature. Metaheuristic algorithms have become popular due to their ability to handle optimization complexity and the continuous influx of text documents. Feature selection is inherently multi-objective, balancing the enhancement of feature relevance, accuracy, and the reduction of redundant features. This research presents a two-fold objective for feature selection. The first objective is to identify the top-ranked features using an ensemble of three multi-univariate filter methods: Information Gain (Infogain), Chi-Square ( $\chi^2$ ), and Analysis of Variance (ANOVA). This aims to maximize feature relevance while minimizing redundancy. The second objective involves reducing the number of selected features and increasing accuracy through a hybrid approach combining Artificial Bee Colony (ABC) and Genetic Algorithms (GA). This hybrid method operates in a wrapper framework to identify the most informative subset of text features. Support Vector Machine (SVM) was employed as the performance evaluator for the proposed model, tested on two high-dimensional multiclass datasets. The experimental results demonstrated that the ensemble filter combined with the ABC+GA hybrid approach is a promising solution for text feature selection, offering superior performance compared to other existing feature selection algorithms.

## KEYWORDS

Metaheuristic algorithms; text classification; multi-univariate filter feature selection; ensemble filter-wrapper techniques

## 1 Introduction

The spontaneous development in the production of electronic data from varied sources (emails, social media sites, online libraries) has brought about an unimaginable advancement in the knowledge domain making it difficult for humans alone to extract adequate information from a variety of disciplines [1]. Since most of the accessible data are text-based, text classification (TC) helps to address



this problem by assigning text documents to different classes [2]. One of the main issues with TC is high-dimensional data where the number of features is far more than the number of samples. This leads to an increase in computational time and complexity requiring more powerful hardware and storage spaces [3]. This has significant effects on the healthcare sector where high dimensional data from medical imaging can overwhelm computational resources, delaying diagnosis. This could also slow down the identification of security threats in cybersecurity. While deep learning models automatically extract relevant features from raw data during the training process, handle large volumes of data, and capture non-linear relationships, traditional feature selection (FS) methods remain important because they produce simpler models, reduce computational complexity when resources are limited, and mitigate overfitting with smaller datasets. Thus, feature selection selects a portion of the relevant features from the original large dataset by removing the irrelevant and noisy features. This will greatly improve the model's efficiency, save time, and increase the overall performance.

Generally, feature selection can be classified into four different techniques namely: filter, wrapper, embedded, and ensemble [4]. In filter methods, irrelevant and unrelated features are eliminated while the best feature subset is selected. Filter methods are not based on classification algorithms for feature selection; this means that this method costs less computing power and usually provides better results with high-dimensional datasets. It uses statistical methods to select and rank features based on features' importance to the target class. Examples are Information Gain (IG), Correlation-based Feature Selection (CFS), Fisher score, Chi-square ( $\chi^2$ ), Analysis of Variance (ANOVA) and minimum Redundancy Maximum Relevance (mRMR). By contrast, wrapper methods use predefined learning algorithms and the predictive performance of a classifier to evaluate the quality of selected feature sets. The wrapper method is subject to considerable computational costs, but finally, it delivers a better result than the filter method. It is widely used with optimization algorithms coupled with machine learning classifiers for feature selection. Examples are feature forward selection, backward feature elimination, and recursive feature elimination. Embedded methods work directly within the learning algorithm to select features based on their importance to the model. It is more effective than wrapper methods because it does not require an iterative evaluation of feature sets. Examples are L1 regularization and Decision Tree (DT). While the individual feature selection method could be unstable, the ensemble method combines the results of various feature selection techniques to form a feature selection ensemble. Individual feature selection methods can be unstable, but the ensemble method combines results from multiple techniques to create a more reliable feature selection ensemble. Hybrid techniques create a different approach by combining the selection of features with other methods, such as dimensionality reduction, transformation, or the Wrapper method. The aim is to leverage the strengths of different techniques to find more suitable features. While ensembles and hybrid methods are data-driven, to improve the selection of features, they differ in some respects. Firstly, ensemble methods make collective decisions based on the combination of multiple feature selection algorithms while hybrid methods integrate feature selection with other methods. Secondly, ensemble methods treat feature selection as a separate step in the machine-learning process and can be applied to a wide range of problems. In contrast, hybrid methods integrate feature selection directly within a specific modelling process, making them more tailored to specific domains and problems. Lastly, ensemble methods combine feature selection techniques using approaches like voting, bagging, stacking, random forest, and boosting [5]. In contrast, hybrid methods combine different feature selection techniques, such as filter and wrapper methods, wrapper and wrapper methods, or wrapper and embedded methods [6].

In recent times, metaheuristic algorithms have been used in solving nondeterministic polynomial time-hard (NP-hard) optimization problems and difficult search problems by allotting a computational method to optimize the problem through the iterative generation process to find the near-best solutions [7]. They have also gained wide popularity in feature selection-related problems of text classification in the following optimization algorithms: Genetic [8], Ant Colony Optimization [9], Particle Swarm Optimization [10], Bat [11], Firefly [12], Gray Wolf Optimizer [13], Gravitational Search Algorithm [14]. They can find global optimality within a limited number of iterations, making the metaheuristic technique suitable for solving large global optimization problems [15].

Recently, pre-trained language models (PLMs) like BERT (Bidirectional Encoder Representations from Transformers), DistilBERT (Distilled BERT), RoBERTa (Robustly Optimized BERT Pretraining Approach), XLNet (Generalized Autoregressive Pretraining for Language Understanding), and XLM (Cross-lingual Language Model) have achieved impressive results in various natural language processing (NLP) tasks. These models are advantageous because they reduce the need for extensive feature engineering and data cleaning. However, since PLMs are designed to be general-purpose, they might not perform as well on tasks within specific domains [16]. A comparison study of Support Vector Machine (SVM) and pre-trained language models (PLMs) for text classification has further demonstrated this point. The literature shows that the SVM classifier performs better than PLMs on domain-specific datasets, such as the 20 Newsgroups and BBC datasets [17].

However, due to the multi-objective nature of feature selection processes, no one method fits all cases in the existing literature [18]. The battle confronting the optimization community is how to get rid of being stuck in local optima and stagnation [19,20]. This study proposes a variation in the search space selection to avoid being stuck and prevent stagnation. It proposes the combination of the strength of ensemble filter methods ( $\chi^2$ , infogain and ANOVA) with the hybrid of metaheuristic algorithms (ABC+GA) for feature selection optimization. Artificial Bee Colony (ABC) is one of the Swarm Intelligence (SI) algorithms, first proposed in 2005 [21]. It models the foraging behaviour of bee colonies. Since its invention, it has drawn much attention for its simplicity and ability to solve real-world optimization problems, such as quadratic assignment [22], sentiment classification [23], and automatic programming [24]. While ABC emphasizes exploration it suffers from exploitation [25,26]. The proposed hybrid is generated by introducing pre-trained model features and weights from the hybridization of three univariate filter feature selection methods ( $\chi^2$ , Infogain, and ANOVA) as input to the ABC model. Thus, the pre-trained model features serve as input to the ABC model, leveraging the knowledge embedded in the pre-trained model, while the weights serve as coefficients during the objective function evaluation to emphasize or de-emphasize specific features based on their importance. The result indicates that the proposed method has the highest fitness score and better precision than other state-of-the-art optimization algorithms for minimal feature subsets. The main contributions of this study are stated as follows:

- An algorithm with a multi-objective function aims to maximize relevance and accuracy while minimizing the number of features and redundancy.
- A similarity-based mutation and random mutation, using pre-trained weights, perturb the mutation process during the employed bee phase. A tournament-induced strategy for onlooker bees adds diversity to the population, preventing premature convergence to local optima and enhancing exploitation.
- An ensemble of multi-univariate filter feature selection methods combined with a hybrid of ABC and GA (e-ABC+GA) is proposed to achieve the algorithm's objectives.

- Experimental comparisons with other advanced optimization algorithms show that the proposed e-ABC+GA achieves the highest fitness scores with a minimal number of features, demonstrating high precision and justifying the objective function.

The rest of this paper is arranged accordingly: Related works are contained in [Section 2](#). A brief description of feature selection methods used in this research is presented in [Section 3](#). [Section 4](#) explains the proposed method. [Section 5](#) describes experimental analysis while [Section 6](#) contains the results and discussion of the findings. [Section 7](#) presents the conclusion and future works.

## 2 Related Works

Researchers have carried out extensive research with FS using ensemble and optimization algorithms for either single or multi-objective related problems. For ensemble algorithms, the following works of literature are discussed: Ensemble Feature Selection using Mutual Information (EFS-MI) combines the subsets using Mutual Information (MI) to reduce the redundancy among the selected features [27], due to the instability of many feature selection methods, an ensemble method has been proposed. This method selects the best subsets of features by combining different subsets that share information between classes and types. The experiments showed a wide range of classification accuracy. Hesitant Fuzzy Sets-Feature Selection (HFS-FS) is another proposed ensemble FS using feature relevancy and feature importance for sentiment classification [28]. The relevancy is calculated through ranking while the importance is estimated through hesitant fuzzy sets. This recorded better classification precision. Another researcher used a heterogeneous approach of filters to generate multiple top k feature candidate rankings with various clustering-based methods using the mean-shift algorithm (EFS-MSCA) [29]. Another researcher proposed ensemble-based filter Feature Selection (EFFS) [30] for recognizing human activity through the selection of features that are robust to the placement of sensors with distinctive activities. This resulted in higher accuracy with fewer features. Other researchers used the VIKOR method as an Ensemble Feature Selection Multi-Criteria Decision-Making (EFS-MCDM) algorithm for feature ranking through the evaluation of several feature selection methods as decision-making criteria [31]. The researchers proposed the EFS-MCDM method, a rank features vector, as an output for users to select a desired number of features. Re-ranking and TOPSIS-based ensemble feature selection (RTEFS) [32] method used a four-stage feature selection strategy ranging from feature extraction, through union subset reranking to a multi-objective genetic algorithm to produce a higher accuracy and F-measure compared to base models. For optimization algorithms, the following works are highlighted: An improved Binary Global Harmony Search algorithm (IBGHS) [33] was developed to tackle problems in feature selection by enhancing the global exploration capacity and improving the speed of convergence. The effectiveness of the proposed method was evaluated using KNN and the results were able to compete favourably when compared with other population-based methods. An improved Binary Sparrow Search Algorithm (iBSSA) [34] was proposed to pick the best features, with increased classification accuracy. KNN, RF, and SVM were used to evaluate the robustness of the iBSSA. The results showed outstanding accuracy and precision compared to other metaheuristic algorithms. A multi-objective Feature Selection with Particle Swarm Optimization (PSOMOFS) [35] was developed for dominance of fuzzy goodness comparison and global particle leader determination. The result of which displayed a better performance in exploration and cost. Artificial Bee Colony Feature Selection (ABCFS) [36] was proposed using ABC as a feature selection method. The results demonstrated that ABCFS improved the classification accuracy of text documents using the SVM classifier on both real and benchmark datasets. A fast multi-objective artificial bee colony feature selection (FMABC-FS) was proposed to

consider both the feature cost as well as classification error as objectives of the algorithm [37]. A novel Contiguous Convolutional Neural Network (CCNN) was proposed, using Differential Evolution (DE) for optimization and named the Evolutionary Contiguous Convolutional Neural Network (ECCNN). Additionally, the researchers introduced a swarm-based Deep Neural Network (DNN) that utilizes Particle Swarm Optimization (PSO), called Swarm DNN. The performance results showed a precision of 88.76% for ECCNN and 87.99% for Swarm DNN on 20 Newsgroup dataset [38].

The summary of the related works are summarized in Tables 1 and 2. This multi-objective helps to obtain improved feature subsets having lesser execution time as compared to other algorithms. While multi-objective algorithms in the past literature have helped to improve performance, the challenge of being trapped in local optima or premature convergence persists [19]. Researchers have advocated upcoming work focusing on combining metaheuristic algorithms [20]. Hence a need to explore combining other algorithms that can tackle the natural occurrence of optimization-related issues and also improve performance.

**Table 1:** Summary of related works on ensemble methods

Ensemble FS methods	Reference	Dataset	Classifier
Ensemble Method using Mutual Information (EFS-MI)	Hoque et al., 2018 [29]	Five gene expressions and two network datasets	KNN, DT, SVM, RF
Hesitant Fuzzy Sets Feature Selection (HFS-FS)	Ansari et al., 2019 [30]	Movie, book, music	NB, SVM
Ensemble Feature Selection with ranking with mean-shift algorithm (EFS-MSCA)	Drotár et al., 2019 [31]	Madelon, LED, XoR, CorrAL-100, high-dimensional madelon	Adaboost, NB
Ensemble-based Filter Feature Selection (EFFS)	Tian et al., 2020 [32]	5 subjects	KNN, SVM
Multi-Criteria Decision Making (EFS-MCDM)	Hashemi et al., 2022 [33]	10 datasets	Not specified
Re-ranking and TOPSIS-based ensemble feature selection (RTEFS)	Fu et al., 2023 [34]	20newsgroup, Reuters	SVM, KNN

**Table 2:** Summary of related works on optimisation methods

Optimisation FS methods	Reference	Dataset	Classifier
An improved Binary Global Harmony Search Algorithm (IBGHS)	Gholami et al., 2020 [35]	18 datasets	KNN
An improved Binary Sparrow Search Algorithm (iBSSA)	Gad et al., 2022 [36]	18 datasets	KNN, SVM, RF

(Continued)

**Table 2 (continued)**

Optimisation FS methods	Reference	Dataset	Classifier
Multi-Objective feature Selection with Particle Swarm Optimization (PSOMOFS)	Hu et al., 2021 [37]	12 datasets	Not specified
Artificial Bee Colony Feature Selection (ABCFS)	Balakumar et al., 2019 [38]	20 Newsgroup	SVM
Fast Multi-Objective Artificial Bee Colony Feature Selection (FMABC-FS)	Wang et al., 2020 [39]	12 datasets	Not specified
Evolutionary Contiguous Convolutionary Neural Network (ECCNN) and Swarm based Deep Neural Network (Swarm DNN)	Prabhakar et al., 2022 [40]	BBC, 20 Newsgroup	CNN, DNN

### 3 Research Background

The research background will be discussed using the ensemble filter feature selection method and Artificial Bee Colony.

#### 3.1 Ensemble Filter Feature Selection Method

Filter techniques can be cost-effective and easy to use because the FS task does not involve any learning model. However, the use of a single filter method is unstable since selection criteria differ from one method to another. The motivation for the ensemble is to improve the performance of a single method to produce the most relevant features for their target classes [39]. The ensemble filter methods are explained below:

Chi-square ( $\chi^2$ ): It is a statistical test used to evaluate the significant association between features and their target classes. The higher the value, the more its relevance to the target class.

$$\chi^2 = \sum \frac{(O_i - E_i)^2}{E_i} \quad (1)$$

where:  $\chi^2$  = chi-square,  $O_i$  = observed value (actual value),  $E_i$  = expected value.

Infogain: It measures the reduction in entropy (uncertainty) about the target variable after observing a feature. Features having higher infogain are termed informative for distinguishing between classes.

$$\text{Infogain}(Z, y) = H(y) - H(y|Z) \quad (2)$$

where:  $Z$  = feature matrix,  $y$  = target vector,  $H(y)$  = target variable entropy.

$H(y|Z)$  = target variable conditional entropy given the feature.

ANOVA: It is a statistical method used to analyze the differences among group means (between-group variance to within-group variance). Features with higher scores are considered more relevant.

$$ANOVA(Z, y) = \frac{\left(\frac{SSB}{k}\right)}{\left(\frac{SSE}{(n-k-1)}\right)} \quad (3)$$

where:  $Z$  = feature matrix,  $y$  = target vector

$SSB$  = sum of squares between groups (variance between groups' means)

$SSE$  = sum of squares within groups (variance within groups' means)

$k$  = number of groups (classes),  $n$  = total number of samples.

### 3.2 Artificial Bee Colony (ABC)

The ABC algorithm was based on a swarm intelligence population optimization algorithm, inspired by bees' intelligent forage behavior. ABC identifies the food source position as a possible solution to the optimisation problem, with the nectar size of each point indicating the quality and suitability of the relevant solution. Three types of bees, namely the employed bee, the onlooker bee and the scout bee, developed food source positions in this population. For the employed bees who manage random searches in a neighboring region according to their parent feed source and exchange information with onlooker bees, half of the total population is set aside. The second half will include onlooker bees which are required to search for better food supply positions with good solutions selected by the amount of information supplied by employed bees. The scout bees are the final group of bees. After a scheduled timeframe, any unimproved candidate solution by employed bees or onlooker bees will be discarded by its employed bee. Looking for a new source randomly in the entire exploration, this employed bee is to be turned into a scout bee. The term "food source" is used within the framework of ABC to depict a candidate solution, this can also be referred to as "individual". The candidate solution has a very good fitness rating if the food source contains high levels of nectar. There are four phases in the standard ABC, i.e., initialization, employed bee, onlooker bee and scout bee. ABC iteratively performs an employed bee, onlooker bee and scout bee phases at the same time as its initialization phase until a termination condition is met. The following are the stages of this process.

Initialization phase: ABC initiates its search process with an initial population. Given that the population contains NFS candidate solutions, each of which is generated according to Eq. (4),

$$x_{ab} = x_b^{high} + rand(0, 1) * (x_b^{high} - x_b^{low}) \quad (4)$$

assuming  $a = 1, 2, \dots, NFS$ ,  $b = 1, 2, \dots, D$ . NFS represents employed bees or onlooker bees' number;  $D$  is the exploration dimension;  $x_b^{high}$  and  $x_b^{low}$  connote the high and low bound of the  $b$ th dimension, respectively. Also, the fitness value for individual candidate solution is depicted as Eq. (5),

$$fitness(x_a) = \begin{cases} \frac{1}{1 + f(x_a)}, & \text{if } (f(x_a) \geq 0) \\ 1 + |f(x_a)|, & \text{otherwise} \end{cases} \quad (5)$$

where:  $fitness(x_a)$  denotes the fitness value of the  $a$ th candidate solution position  $x_a$  and  $f(x_a)$  stands for the objective function value of the food source position  $x_a$  for the optimization problem.

Employed bee phase: Employed bees are charged with finding new food sources throughout a wide search area at this stage as depicted by Eq. (6),

$$v_{a,b} = x_{a,b} + \phi_{a,b} * (x_{a,b} - x_{r,b}) \quad (6)$$

where:  $V_a = (v_{a,1}, v_{a,2}, \dots, v_{a,D})$  is the new candidate solution corresponding to the old candidate solution  $X_a$  and  $X_r$  represents a randomly selected candidate solution from the separate population aside  $X_a$ .  $\Phi_{a,b}$  represents an equally spaced number ranging from  $[-1, 1]$  and  $b$  is randomly picked from  $\{1, 2, \dots, D\}$ . Should the fitness value of  $v_a$  be higher than  $x_a$ ,  $x_a$  is swapped with  $v_a$ , while the counter for the repeated unsuccessful updates of the candidate solution position  $x_a$  is returned to 0. If not,  $x_a$  is retained to continue to the subsequent generation and the increment is increased by 1. It is done by a greedy selection method.

Onlooker bee phase: The onlooker bee phase performs a thorough exploration around engaging candidate solution. This search behaviour is exploring the neighbour of solution candidates (local search). Each food source is covered by a selection probability, this can be shown in Eq. (7),

$$p(x_a) = \frac{\text{fitness}(x_a)}{\sum_{b=1}^{NFS} \text{fitness}(x_a)} \quad (7)$$

where:  $p(x_a)$  represents the selection probability. The more the fitness value, the greater the selection probability with a chance of an engaging candidate solution being selected several times. After deciding the candidate solution to be picked, new food sources are produced by Eq. (6) in line with the employed bee phase.

Scout bee phase: This phase's objective is to avoid population inertia, while concurrently developing new food sources. To keep track of the frequency of sources not updated, a counter is set for each food resource. Should its counter value exceed the threshold, the employed bee abandons the candidate solution position with the maximum counter value and turns to a scout bee to commence the search for a new candidate solution through the path of Eq. (4). Hence, once the new candidate solution has been updated, its increment is adjusted to zero, and the cycle of search begins afresh.

It is on this premise of ensemble multi-univariate filter feature selection and hybrid of meta-heuristic algorithm for improved accuracy, feature relevance, and reducing number of features that the proposed model is based on.

#### 4 The Proposed Method

The high dimensionality of the feature space increases computational complexity and processing time, which is detrimental to any machine-learning algorithm. Therefore, to enhance performance, the feature space dimension as well as the computational complexity of the machine learning have to be reduced. To achieve this, an ensemble of multi-univariate filter feature selection methods (Chi<sup>2</sup>, infogain and ANOVA) was carried out through concatenated weighted voting as a first stage process in the feature selection to remove noisy and irrelevant features as demonstrated in our previous work [40]. The choice of (Chi<sup>2</sup>, ANOVA and infogain) filter FS methods is based on different metrics, and combining these methods works better than using just one. This approach has proven effective in text classification applications, providing a unique perspective on feature importance. ABC's ability to solve real-world optimization problems and its ease of implementation make it a valuable optimization algorithm.

This will generate features with improved relevance and reduced redundancy and lead to unique and top-ranked relevant features as output for the next stage. This could be depicted with Algorithm 1.



**Algorithm 1:** Pseudocode for the ensemble of filter feature selection methods

---

```

1. Clean Datasets
   FOR each dataset in [20_newsgroup, 17_newsgroup]:
       Remove headers, punctuations, stopwords, special characters
       Convert capital letters to lowercase
2. Split Datasets
   Split dataset into train, test, validate subsets
   x_train, x_test, x_validate = texts
   y_train, y_test, y_validate = categories
3. Encode Categories
   y_train = label_encoder.fit_transform(y_train)
   y_test = label_encoder.transform(y_test)
   y_validate = label_encoder.transform(y_validate)
4. Process Texts
   FOR each text in x_train, x_test, x_validate:
       Tokenize text
       Lemmatize text
       Convert text to TF-IDF format
   features_count = 106105
   samples_count = 11307
5. Trim Rare Terms
   Remove terms with Document Frequency (DF) < 3
   features_count = 28551
6. Feature Selection
   chi2_scores = chi2(x_train, y_train)
   anova_scores = ANOVA(x_train, y_train)
   infogain_scores = infogain(x_train, y_train)
7. Concatenate Scores
   filtered_scores = concatenate(chi2_scores, anova_scores, infogain_scores)
8. Select Unique Features
   unique_features = select_unique(filtered_scores)
9. Count Frequency
   frequency_count = count_frequency(unique_features)
10. Average Multiple Scores
   averaged_scores = average_scores(unique_features, frequency_count)
11. Select Features by Frequency
   selected_features = []
   FOR feature in unique_features:
       IF frequency_count[feature] >= 2:
           selected_features.append(feature)
       ELSE IF feature_has_high_discriminative_score(feature):
           selected_features.append(feature)
12. Create MUNIFES
   MUNIFES = selected_features

```

---

(Continued)

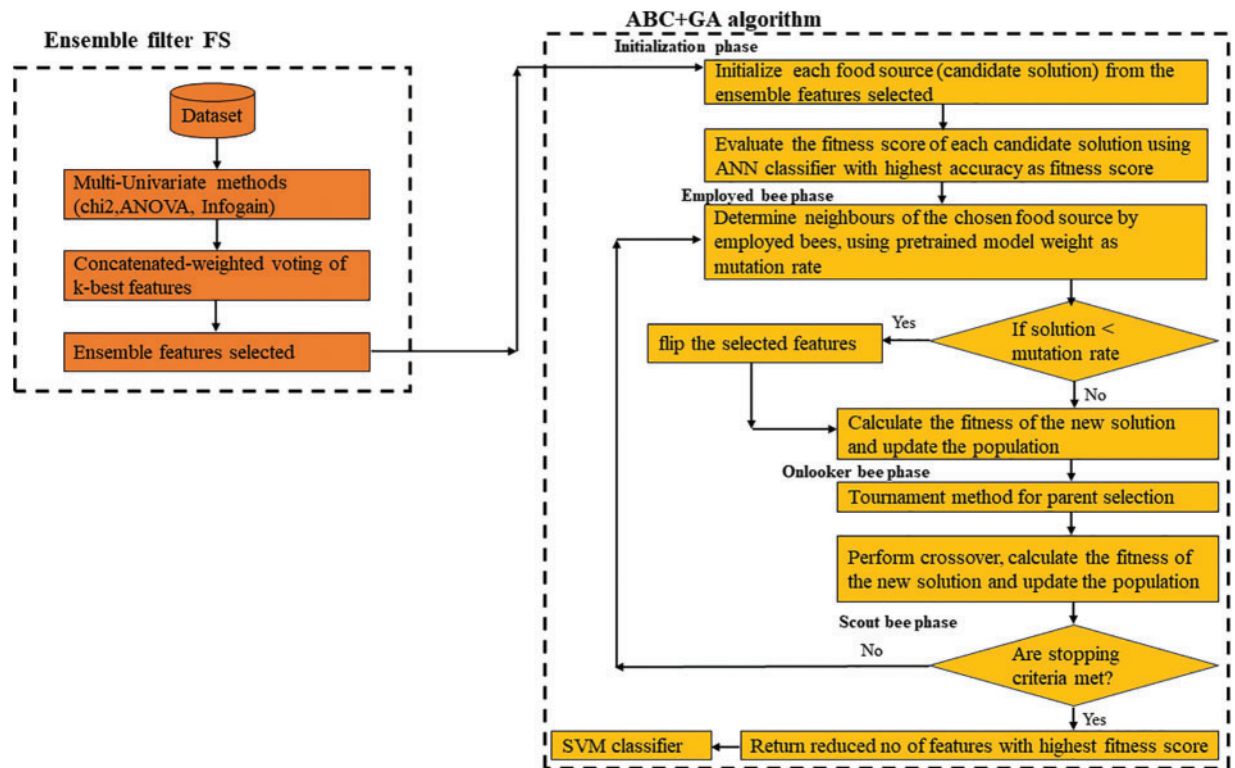
**Algorithm 1 (continued)**

13. Repeat for Different FS Sizes

FOR  $fs\_size$  in range(100, 1000, 100):

Repeat Steps 7 to 12

From the pseudocode, the top I-ranked relevant features were selected using three feature selection methods over 100 to 1000 iterations, in intervals of 100. Unique features from these selections were then combined into a single array. The selected features become the input to the ABC+GA algorithm part. The framework for the proposed feature selection method is represented in Fig. 1.



**Figure 1:** e-ABC+GA model

From Fig. 1, the reduced feature set selected becomes an input to the ABC+GA algorithm to obtain the nearest best feature subset. Since the FS problem is binary by nature, Artificial Neural Network (ANN) was used as an evaluation classifier for each of the candidate solution subsets due to its effectiveness in various domains including text classification [41].

Initialization phase: From the standard ABC equations discussed above, the algorithm starts producing a random population of food sources with the ensemble selected features and revises its population for a global exploration throughout iterations until the stop criteria are met. Each food source has a fixed [0,1] range, where each element indicates the presence (1) or absence (0) in its selection. Evaluation of each candidate solution was carried out by ANN classifier. The problem is to discover the best subset of features that maximizes a predefined fitness function while the fitness function evaluates the performance of the selected features using a pre-trained model. The evaluation criteria are based on the fitness of the feature subset. This research adopts a maximization objective

function of maximizing accuracy [42,43]. It searches for the candidate solution, while maximizing accuracy (correctly classifying instances), that will yield the highest fitness score with a smaller number of features.

Employee bee phase: In this phase, the mutation ability of the Genetic algorithm is introduced to strengthen global exploration through the pre-trained weights from the first layer of a neural network to perturb the mutation operation and serve as the mutation rate. The dynamic modification of the mutation rate in the optimization process, due to the variation in the weights of the pre-trained model features, helps to avoid stagnation and premature convergence, improving the algorithm's performance.

This is expressed in Algorithm 2 containing the pseudocode of similarity-based mutation.

---

**Algorithm 2:** Pseudocode of the similarity-based solution

---

```

first_layer_weights = pretrained_model.layers [0].get_weights() [0]
functional_weights = first_layer_weights [0]
function calculate_mutation_rate(solution, functional_weights):
    solution = convert_to_float32(solution)
    functional_weights = convert_to_float32(functional_weights)
    similarity = dot_product(solution, functional_weights)
    mutation_rate = similarity
    return mutation_rate

```

---

Onlooker bee phase: Tournament selection is a mechanism for selecting individuals for reproduction based on a tournament-style competition and the winner from the contestant individuals is chosen to proceed for crossover. This supports diversification. The selection of the fittest bee in each tournament promotes exploitation. Hence, preventing the features from being trapped or premature convergence. The pseudocode of the tournament selection is outlined in Algorithm 3.

---

**Algorithm 3:** Pseudocode for the tournament selection

---

```

num_onlooker_bees = num_scouts
tournament_size = 5
# Onlooker Bee Phase
for i in range(num_scouts):
    parent1 = tournament_selection(population, fitness_scores, tournament_size)
    parent2 = tournament_selection(population, fitness_scores, tournament_size)
    # Crossover
    min_length = min(len(parent1), len(parent2))
    crossover_point = random.randint(1, min_length - 1)
    # Perform Crossover
    new_solution = np.concatenate((parent1[:crossover_point], parent2[crossover_point:]))
    # Pad the new solution if it's shorter than num_features
    if len(new_solution) < num_features:
        new_solution = np.concatenate((new_solution, np.zeros(num_features -
            len(new_solution))))
# Tournament Selection Function
def tournament_selection(population, fitness_scores, tournament_size):

```

---

(Continued)

**Algorithm 3 (continued)**


---

```

tournament_indices = random.sample(range(len(population)), tournament_size)
tournament_fitness = [fitness_scores[i] for i in tournament_indices]
selected_index = tournament_indices[np.argmax(tournament_fitness)]
return population[selected_index]

```

---

**5 Experimental Analysis****5.1 Datasets**

The 20 Newsgroups dataset, a popular standard collection, comprises 18,846 documents gathered from 20 different newsgroups, and the 17 Newsgroups dataset is a variant of the 20 Newsgroup with 16,075 documents. The dataset is divided into training, validation, and test sets. A validation set is for hyperparameter tuning and decision-making during training. A summary of the attributes of the datasets is shown in [Table 3](#).

**Table 3: Benchmark datasets**

Datasets	Classes number	Features number	Documents number
20 Newsgroup	20	106,105	18,846
*17 Newsgroup	17	105,657	16,075

Note: \*17 Newsgroup excepts soc.religion.christian, misc.forsale and alt.atheism classes.

**5.2 Measurement Criteria**

In this study, fitness score, number of selected features, precision, recall, and F1-score are the metrics used to evaluate the performance of e-ABC+GA.

$$Precision = \frac{TP}{TP + FP} \quad (8)$$

$$Recall = \frac{TP}{TP + FN} \quad (9)$$

$$F1 - score = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (10)$$

Given  $TP$  as True Positives (correctly predicted positives)

$TN$  as True Negatives (correctly predicted negatives)

$FP$  as False Positives (incorrectly predicted positives)

$FN$  as False Negatives (incorrectly predicted negatives).

**5.3 Other Metaheuristic Algorithms for Comparison**

The metaheuristic-based algorithms are typically categorized into population-based and local search (single solution) algorithms. Population-based algorithms analyze multiple areas of the search space simultaneously, aiming to enhance them collectively. This approach helps maintain diversity within the population and prevents solutions from becoming trapped in local optima. Population-based algorithms are further subdivided into Evolutionary-based techniques, which include examples

such as Genetic Algorithm (GA) and Differential Evolution (DE), and Swarm-intelligence Algorithms like Particle Swarm Optimization (PSO), Artificial Bee Colony Optimization (ABC), Ant Colony Optimization (ACO), and Bat Algorithm (BA). For this evaluation, both evolutionary-based techniques (DE and GA) and swarm-intelligence algorithms (PSO and ABC) will be utilized for comparison with the proposed method.

## 6 Results and Discussion

### 6.1 Performance Analysis of e-ABC+GA

To verify the performance of e-ABC+GA with other metaheuristics algorithms, fitness score, number of selected features, class-specific metrics of precision, recall, and F1-score were used for comparison. The statistical significance of the proposed method was also calculated.

#### 6.1.1 Fitness Scores Comparison

The fitness score comparison is shown in Figs. 2 and 3 for 20 Newsgroup and 17 Newsgroup.

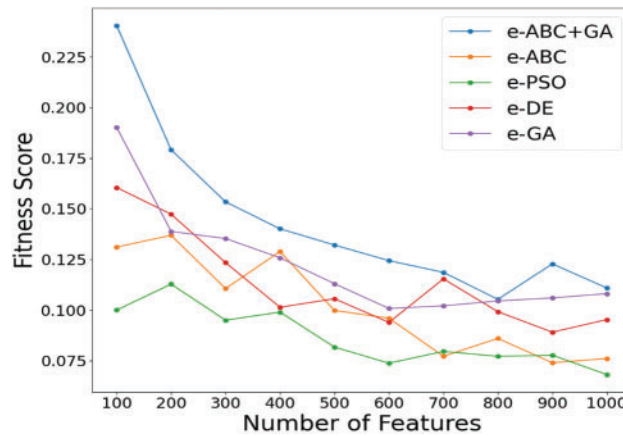


Figure 2: Fitness score comparison for 20 Newsgroup

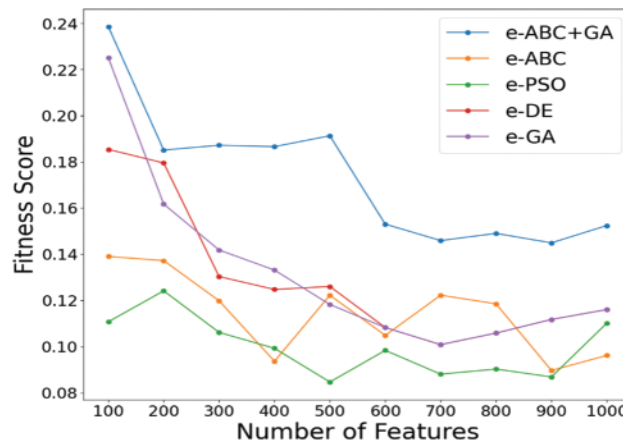


Figure 3: Fitness score comparison for 17 Newsgroup

It can be seen from Figs. 2 and 3 that the fitness scores of e-ABC+GA are higher than those of other metaheuristic algorithms that use the same ensemble features as input across iterations, ranging from 100 to 1000 for both datasets. This justifies the maximization objective function.

### 6.1.2 Number of Selected Features Comparison

The performance of e-ABC+GA is evaluated by the features selected number in comparison with the ensemble features as shown in Tables 4 and 5.

**Table 4:** Percentage of selected features number reduction for 20 Newsgroup

Methods	100	200	300	400	500	600	700	800	900	1000	Average %
Ensemble	103	212	321	426	485	562	644	721	803	882	
e-ABC+GA	66	114	142	210	233	259	320	368	404	441	
% Reduction	64.1	53.8	44.2	49.3	48.0	46.1	49.7	51.0	50.3	50.0	50.7

**Table 5:** Percentage of selected features number reduction for 17 Newsgroup

Methods	100	200	300	400	500	600	700	800	900	1000	Average %
Ensemble	105	209	319	396	459	540	625	702	774	856	
e-ABC+GA	57	108	134	195	231	249	311	340	361	435	
% Reduction	54.3	51.7	42.0	49.2	50.3	46.1	49.8	48.4	46.6	50.8	48.9

Tables 4 and 5 show across iterations (100 to 1000) that there is a 50.7% and 48.9% reduction on average in the number of selected features for 20 Newsgroup and 17 Newsgroup when compared with the selected features from ensemble input features. This has revealed that e-ABC+GA method has improved the efficiency of the classification by reducing model complexity. This will in turn enhance the performance of the model.

### 6.1.3 Class-Specific Metrics of e-ABC+GA with Existing Methods

To understand the behavior of the proposed method across the classes, it was compared with an existing method using precision, recall, and F-measure. Balakumar and Mohan [36] used ABC as a feature selection technique to classify 20 Newsgroup documents based on their content on the SVM classifier (ABCFS). The comparison is shown in Table 6 between ABCFS and e-ABC+GA.

**Table 6:** Performance of e-ABC+GA across classes on SVM classifier–20 Newsgroups

Classes	Algorithms	Precision	Recall	F-measure
rec.sport.hockey	*ABCFS	0.668	0.634	0.651
	e-ABC+GA	<b>0.78</b>	<b>0.73</b>	<b>0.75</b>
rec.motorcycles	*ABCFS	0.61	<b>0.624</b>	0.617
	e-ABC+GA	<b>0.96</b>	0.6	<b>0.74</b>
rec.sport.baseball	*ABCFS	0.715	<b>0.623</b>	<b>0.666</b>

(Continued)

**Table 6 (continued)**

Classes	Algorithms	Precision	Recall	F-measure
rec.autos	e-ABC+GA	<b>0.94</b>	0.44	0.59
	*ABCFS	0.798	<b>0.723</b>	<b>0.759</b>
sci.crypt	e-ABC+GA	<b>0.83</b>	0.49	0.62
	*ABCFS	0.611	0.625	0.618
sci.med	e-ABC+GA	<b>0.91</b>	<b>0.63</b>	<b>0.74</b>
	*ABCFS	<b>0.671</b>	<b>0.682</b>	<b>0.676</b>
comp.windows.x	e-ABC+GA	0.56	0.41	0.47
	*ABCFS	<b>0.703</b>	<b>0.764</b>	<b>0.732</b>
sci.space	e-ABC+GA	0.53	0.52	0.53
	*ABCFS	0.758	<b>0.744</b>	<b>0.751</b>
	e-ABC+GA	<b>0.8</b>	0.63	0.71

Note: \*Adopted from Balakumar et al. [36].

It could be observed from [Table 6](#) comparison between the ABCFS and proposed e-ABC+GA on eight classes of 20 Newsgroup, that despite the 50.7% reduction in the number of features of e-ABC+GA, it could compete favorably with ABCFS in many of the classes recording a 96% precision in rec.motorcycles class. However, there were trade-offs in sci.med and comp.windows.x classes where ABCFS outperformed e-ABC+GA.

#### 6.1.4 Statistical Significance

To ascertain the strength of the proposed model, a statistical test ( $t$ -test) that compares the results of values across intervals is employed using 95% confidence level based on the  $p$ -values was used. [Tables 7](#) and [8](#) show the details of the comparison.

**Table 7:**  $t$ -test comparison between e-ABC+GA and other algorithms–20 Newsgroup

Methods	$t$ -statistic	$p$ -value	Significant
e-ABC+GA AND e-ABC	4.882307467	0.000868607	+
e-ABC+GA AND e-PSO	5.658721965	0.000310139	+
e-ABC+GA AND e-DE	4.390694912	0.001743791	+
e-ABC+GA AND e-GA	4.232298673	0.002199138	+

It could be seen that the proposed method has statistical significance with 95% confidence across all other metaheuristic algorithms on both datasets. This further strengthens the superiority of the proposed method.

**Table 8:** *t*-test comparison between e-ABC+GA and other algorithm–17 Newsgroup

Methods	<i>t</i> -statistic	<i>p</i> -value	Significant
e-ABC+GA AND e-ABC	7.704327706	0.0000299	+
e-ABC+GA AND e-PSO	8.664492747	0.0000116	+
e-ABC+GA AND e-DE	8.175781319	0.0000186	+
e-ABC+GA AND e-GA	5.286102676	0.0005031	+

## 6.2 Discussion

The high dimensionality of the 20 Newsgroups dataset increases computational complexity and processing time. To enhance performance, we used an ensemble of Chi<sup>2</sup>, ANOVA, and Information Gain methods to remove noisy and irrelevant features, reducing the feature space. The resulting features were then input into a hybrid of ABC (Artificial Bee Colony) and GA (Genetic Algorithm) for further optimization. This hybrid approach focuses on maximizing a fitness function that directly measures solution quality. The result is a reduced feature set with the highest fitness score. It was seen that the proposed method e-ABC+GA produced the highest fitness scores across iterations ranging from 100 to 1000 at the interval of 100 across 20 Newsgroup and 17 Newsgroup datasets. This position was confirmed with other metaheuristic algorithms (swarm intelligence–ABC, PSO, and evolutionary–GA, DE) which demonstrated the superiority of e-ABC+GA. It was also confirmed that the proposed method produced an average of 50.7% and 48.9% respective reductions in the number of features selected compared with the input features (ensemble) in 20 Newsgroups and 17 Newsgroups. The behaviour of the proposed model across classes was tested on precision, recall and F1-score on the SVM classifier, despite the 50.7% and 48.9% reductions in the number of selected features, it competed favorably with 96% precision on 20 Newsgroup datasets. e-ABC+GA was statistically significant with 95% confidence using a *t*-test compared to other metaheuristic algorithms. All these have justified the improvement recorded by the proposed method (e-ABC+GA). Generally, optimization approaches to feature selection consume a lot of system memory due to the many iterations needed to find near-optimal solutions.

## 7 Conclusion and Future Works

The research work initiates a new approach to address the text feature selection problem using text classification which hybridizes an ensemble of multi-univariate filter feature selection methods (Chi<sup>2</sup>, infogain and ANOVA) with wrapper approaches of ABC+GA using SVM classifier. This method is termed e-ABC+GA. The top-I ranked features selected from the concatenation of the ensemble features of Chi<sup>2</sup>, infogain and ANOVA were passed as input to the wrapper ABC. This ensemble selection through weighted voting was to eliminate irrelevant and noisy features and reduce the local and global search spaces. The perturbation introduced through the variation in the pre-trained model weights at the employed bee phase with GA tournament selection and crossover in the onlooker bee phase has strengthened the local search from stagnation. This has enabled the ABC+GA to generate an improved subset of important features from top relevant selected features. The proposed method has achieved a better performance than other swarm intelligence and evolutionary algorithms in the literature using fitness score as a metric for evaluation. The comparison with existing methods using F-measure, recall, and precision recorded 96% precision on 20 Newsgroups has confirmed the efficiency of e-ABC+GA as a favourable approach for addressing feature selection-related problems



in text classification. However, the proposed method has only been verified using the SVM classifier. In the future, verification with other classifiers is still needed. Additionally, the recall metric needs improvement for better performance across all classes.

**Acknowledgement:** The authors thank the Universiti Sains Malaysia (USM) and the School of Computer Sciences, USM for sponsoring the manuscript. The authors are also grateful to the editor, and anonymous reviewers for their comments and suggestions to enhance the quality of the paper.

**Funding Statement:** This study was supported by Universiti Sains Malaysia (USM) and School of Computer Sciences, USM.

**Author Contributions:** The authors confirm their contribution to the paper as follows: study conception and design: Oluwaseun Peter Ige, Keng Hoon Gan; data collection: Oluwaseun Peter Ige; analysis and interpretation of results: Oluwaseun Peter Ige, Keng Hoon Gan; draft manuscript preparation: Oluwaseun Peter Ige. All authors reviewed the results and approved the final version of the manuscript.

**Availability of Data and Materials:** The dataset is available for download from <https://www.kaggle.com/datasets/crawford/20-newsgroups> (accessed on 26 June 2024), <https://ndownloader.figshare.com/files/5975967> (accessed on 25 June 2024) or directly from sklearn library “from sklearn.datasets import fetch\_20newsgroups” into the python code.

**Ethics Approval:** Not applicable.

**Conflicts of Interest:** The authors declare that they have no conflicts of interest to report regarding the present study.

## References

1. Garba Sharifai A, Zainol Z. The correlation-based redundancy multiple-filter approach for gene selection. *Int J Data Min Bioinform.* 2020;23(1):62–78. doi:10.1504/IJDMB.2020.105437.
2. Demoulin NTM, Coussement K. Acceptance of text-mining systems: the signaling role of information quality. *Inform Manage.* 2020;57(1):103120. doi:10.1016/j.im.2018.10.006.
3. Jia W, Sun M, Lian J, Hou S. Feature dimensionality reduction: a review. *Complex Intell Syst.* 2022;8(3):2663–93. doi:10.1007/s40747-021-00637-x.
4. Abasabadi S, Nematzadeh H, Motameni H, Akbari E. Automatic ensemble feature selection using fast non-dominated sorting. *Inf Syst.* 2021;100(2):101760. doi:10.1016/j.is.2021.101760.
5. Mohammed A, Kora R. A comprehensive review on ensemble deep learning: opportunities and challenges. *J King Saud Univ–Comput Informati Sci.* 2023;35(2):757–74. doi:10.1016/j.jksuci.2023.01.014.
6. Elemam T, Elshrkawey M. A highly discriminative hybrid feature selection algorithm for cancer diagnosis. *Sci World J.* 2022;2022(3):1–15. doi:10.1155/2022/1056490.
7. Peres F, Castelli M. Combinatorial optimization problems and metaheuristics: review, challenges, design, and development. *Appl Sci.* 2021;11(14):6449. doi:10.3390/app11146449.
8. Katoch S, Chauhan SS, Kumar V. A review on genetic algorithm: past, present, and future. *Multimed Tools Appl.* 2021;80(5):8091–126. doi:10.1007/s11042-020-10139-6.
9. Mayet AM, Ijyas VPT, Bhutto JK, Guerrero JWG, Shukla NK, Eftekhari-Zadeh E, et al. Using ant colony optimization as a method for selecting features to improve the accuracy of measuring the thickness of scale in an intelligent control system. *Processes.* 2023;11(6):1621. doi:10.3390/pr11061621.

10. Kumar V, Singh Aydav PS, Minz S. Multi-view ensemble learning using multi-objective particle swarm optimization for high dimensional data classification. *J King Saud Univ-Comput Inform Sci.* 2022;34(10):8523–37. doi:10.1016/j.jksuci.2021.08.029.
11. Al-Betar MA, Alomari OA, Abu-Romman SM. A TRIZ-inspired bat algorithm for gene selection in cancer classification. *Genomics.* 2020;112(1):114–26. doi:10.1016/j.ygeno.2019.09.015.
12. Bazi S, Benzid R, Bazi Y, Al Rahhal MM. A fast firefly algorithm for function optimization: application to the control of bldc motor. *Sensors.* 2021;21(16):5267. doi:10.3390/s21165267.
13. Alyasiri OM, Cheah YN, Abasi AK. Hybrid filter-wrapper text feature selection technique for text classification. In: *International Conference on Communication and Information Technology, ICICT 2021, 2021; Basrah, Iraq: Institute of Electrical and Electronics Engineers Inc.;* p. 80–6. doi:10.1109/ICICT52195.2021.9567898
14. Wang Y, Gao S, Zhou M, Yu Y. A multi-layered gravitational search algorithm for function optimization and real-world problems. *IEEE/CAA J Automatica Sinica.* 2021;8(1):94–109. doi:10.1109/JAS.2020.1003462.
15. Seyyedabbasi A. A reinforcement learning-based metaheuristic algorithm for solving global optimization problems. *Adv Eng Softw.* 2023;178(3):103411. doi:10.1016/j.advengsoft.2023.103411.
16. Arslan Y, Allix K, Veiber L, Lothritz C, Bissyandé TF, Klein J, et al. A comparison of pre-trained language models for multi-class text classification in the financial domain. In: *The Web Conference 2021-Companion of the World Wide Web Conference, 2021; Ljubljana, Slovenia: Association for Computing Machinery, Inc.;* p. 260–8. doi:10.1145/3442442.3451375.
17. Wahba Y, Madhavji N, Steinbacher J. A comparison of SVM against pre-trained language models (PLMs) for text classification tasks. In: *Nicosia G et al., editors. Machine learning, optimization, and data science. Cham Switzerland: Springer; vol. 13811, 2023.* doi:10.1007/978-3-031-25891-6\_23.
18. Rahimi I, Gandomi AH, Chen F, Mezura-Montes E. A review on constraint handling techniques for Population-based algorithms: from single-objective to multi-objective optimization. *Netherlands: Springer; 2023.* doi:10.1007/s11831-022-09859-9.
19. Dokeroglu T, Deniz A, Kiziloz HE. A comprehensive survey on recent metaheuristics for feature selection. *Neurocomputing.* 2022;494(13):269–96. doi:10.1016/j.neucom.2022.04.083. Elsevier B.V..
20. Akinola OO, Ezugwu AE, Agushaka JO, Zitar RA, Abualigah L. Multiclass feature selection with metaheuristic optimization algorithms: a review. *London: Springer; 2022.* doi:10.1007/s00521-022-07705-4.
21. Karaboga D, Gorkemli B, Ozturk C, Karaboga N. A comprehensive survey: artificial bee colony (ABC) algorithm and applications. *Artif Intell Rev.* 2014;42(1):21–57. doi:10.1007/s10462-012-9328-0.
22. Akay B, Karaboga D, Gorkemli B, Kaya E. A survey on the Artificial Bee Colony algorithm variants for binary, integer and mixed integer programming problems. *Appl Soft Comput.* 2021;106(3):107351. doi:10.1016/j.asoc.2021.107351.
23. Zhang M, Palade V, Wang Y, Ji Z. Attention-based word embeddings using Artificial Bee Colony algorithm for aspect-level sentiment classification. *Inf Sci.* 2021;545(1):713–38. doi:10.1016/j.ins.2020.09.038.
24. Nekoei M, Moghaddas SA, Mohammadi Golafshani E, Gandomi AH. Introduction of ABCEP as an automatic programming method. *Inf Sci.* 2021;545:575–94. doi:10.1016/j.ins.2020.09.020.
25. Zhou X, Lu J, Huang J, Zhong M, Wang M. Enhancing artificial bee colony algorithm with multi-elite guidance. *Inf Sci.* 2021;543(2):242–58. doi:10.1016/j.ins.2020.07.037.
26. Bajer D, Zorić B. An effective refined artificial bee colony algorithm for numerical optimisation. *Inf Sci.* 2019;504(1):221–75. doi:10.1016/j.ins.2019.07.022.
27. Hoque N, Singh M, Bhattacharyya DK. EFS-MI: an ensemble feature selection method for classification. *Complex Intell Syst.* 2018;4(2):105–18. doi:10.1007/s40747-017-0060-x.
28. Ansari G, Ahmad T, Doja MN. Ensemble of feature ranking methods using hesitant fuzzy sets for sentiment classification. *Int J Mach Learn Comput.* 2019;9(5):599–608. doi:10.18178/ijmlc.2019.9.5.846.

29. Drotár P, Gazda M, Vokorokos L. Ensemble feature selection using election methods and ranker clustering. *Inf Sci.* 2019;480(12):365–80. doi:10.1016/j.ins.2018.12.033.
30. Tian Y, Zhang J, Wang J, Ma J. Multiple feature fusion by hierarchical ensemble with mean target-neighbor distance for music emotion recognition. In: *Proceedings of the 8th International Conference on Multimedia Information Retrieval, 2017*; Bucharest, Romania: Association for Computing Machinery, Inc.; pp. 356–69. doi:10.1145/3078971.3079017.
31. Hashemi A, Dowlatshahi MB, Nezamabadi-pour H. Ensemble of feature selection algorithms: a multi-criteria decision-making approach. *Int J Mach Learn Cybern.* 2022;13(1):49–69. doi:10.1007/s13042-021-01347-z.
32. Fu G, Li B, Yang Y, Li C. Re-ranking and TOPSIS-based ensemble feature selection with multi-stage aggregation for text categorization. *Pattern Recognit Lett.* 2023;168(3):47–56. doi:10.1016/j.patrec.2023.02.027.
33. Gholami J, Pourpanah F, Wang X. Feature selection based on improved binary global harmony search for data classification. *Appl Soft Comput J.* 2020;93(12):106402. doi:10.1016/j.asoc.2020.106402.
34. Gad AG, Sallam KM, Chakraborty RK, Ryan MJ, Abohany AA. An improved binary sparrow search algorithm for feature selection in data classification. *Neural Comput Appl.* 2022;34(18):15705–52. doi:10.1007/s00521-022-07203-7.
35. Hu Y, Zhang Y, Gong D. Multiobjective particle swarm optimization for feature selection with fuzzy cost. *IEEE Trans Cybern.* 2021;51(2):874–88. doi:10.1109/TCYB.2020.3015756.
36. Balakumar J, Mohan SV. Artificial bee colony algorithm for feature selection and improved support vector machine for text classification. *Inf Discov Deliv.* 2019;47(3):154–70. doi:10.1108/IDD-09-2018-0045.
37. Wang XH, Zhang Y, Sun XY, Wang YL, Du CH. Multi-objective feature selection based on artificial bee colony: an acceleration approach with variable sample size. *Appl Soft Comput J.* 2020;88(4):106041. doi:10.1016/j.asoc.2019.106041.
38. Prabhakar SK, Rajaguru H, So K, Won DO. A framework for text classification using evolutionary contiguous convolutional neural network and swarm based deep neural network. *Front Comput Neurosci.* 2022;16:900885. doi:10.3389/fncom.2022.900885.
39. Hashemi A, Bagher Dowlatshahi M, Nezamabadi-pour H. A pareto-based ensemble of feature selection algorithms. *Expert Syst Appl.* 2021;180(5):115130. doi:10.1016/j.eswa.2021.115130.
40. Ige O, Gan KH. Ensemble feature selection using weighted concatenated voting for text classification. *J Niger Soc Phys Sci.* 2024;1823. doi:10.46481/jnsps.2024.1823.
41. Hartmann J, Huppertz J, Schamp C, Heitmann M. Comparing automated text classification methods. *Int J Res Mark.* 2019;36(1):20–38. doi:10.1016/j.ijresmar.2018.09.009.
42. Too J, Abdullah AR, Saad NM, Tee W. EMG feature selection and classification using a Pbest-guide binary particle swarm optimization. *Computation.* 2019;7(1):12. doi:10.3390/computation7010012.
43. Tubishat M, Tubishat M, Ja'afar S, Alswaiti M, Mirjalili S, Idris N, et al. Dynamic salp swarm algorithm for feature selection. *Expert Syst Appl.* 2021;164(1):113873. doi:10.1016/j.eswa.2020.113873.