IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 1 March 2024, accepted 3 April 2024, date of publication 18 April 2024, date of current version 8 May 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3390684

APPLIED RESEARCH

Feature Subset Selection for High-Dimensional, Low Sampling Size Data Classification Using Ensemble Feature Selection With a Wrapper-Based Search

ASHIS KUMAR MANDAL[1,2], MD. NADIM[1,2], HASI SAHA[2], TANGINA SULTANA[3,4], MD. DELOWAR HOSSAIN[2,4], AND EUI-NAM HUH[4], (Member, IEEE)

[1]Department of Computer Science, University of Saskatchewan, Saskatoon, SK S7N 5A2, Canada
[2]Department of Computer Science and Engineering, Hajee Mohammad Danesh Science and Technology University, Dinajpur 5200, Bangladesh
[3]Department of Electronics and Communication Engineering, Hajee Mohammad Danesh Science and Technology University, Dinajpur 5200, Bangladesh
[4]Department of Computer Science and Engineering, Kyung Hee University, Yongin 17104, South Korea

Corresponding author: Eui-Nam Huh (johnhuh@khu.ac.kr)

This work was supported in part by the Institute of Information and Communications Technology Planning and Evaluation (IITP) grant funded by Korea Government (MSIT) (Artificial Intelligence Innovation Hub) under Grant 2021-0-02068; and in part by the Artificial Intelligence Convergence Innovation Human Resources Development, Kyung Hee University, under Grant RS-2022-00155911.

ABSTRACT The identification of suitable feature subsets from High-Dimensional Low-Sample-Size (HDLSS) data is of paramount importance because this dataset often contains numerous redundant and irrelevant features, leading to poor classification performance. However, the selection of an optimal feature subset from a vast feature space creates a significant computational challenge. In the domain of HDLSS data, conventional feature selection methods often face challenges in achieving a balance between reducing the number of features and preserving high classification accuracy. Addressing these issues, the study introduces an effective framework that employs a filter and wrapper-based strategy specifically designed to address the classification challenges inherent in HDLSS data. The framework adopts a multi-step approach where ensemble feature selection integrates five filter ranking approaches: Chi-square (χ^2), Gini index (GI), F-score, Mutual Information (MI), and Symmetric uncertainty (SU) to identify the top-ranking features. In the subsequent stage, a wrapper-based search method is utilized, which employs the Differential Evaluation (DE) metaheuristic algorithm as the search strategy. The fitness of feature subsets during this search is assessed based on a weighted combination of the error rate of the Support Vector Machine (SVM) classifier and the ratio of feature cardinality. The datasets, after undergoing dimensionality reduction, are then utilized to construct classification models using SVM, K-Nearest Neighbors (KNN), and Logistic Regression (LR). The approach was evaluated on 13 HDLSS datasets to assess its efficacy in selecting appropriate feature subsets and improving Classification Accuracy (ACC) analog with Area Under the Curve (AUC). Results show that the proposed ensemble with wrapper-based approach produces a smaller number of features (ranging between 2 and 9 for all datasets), while maintaining a commendable average AUC and ACC (between 98% and 100%). The comparative analysis reveals that the proposed method surpasses both ensemble feature selection and non-feature selection approaches in terms of feature reduction and ACC. Additionally, when compared to various other state-of-the-art methods, this approach demonstrates commendable performance.

INDEX TERMS Classification, differential evaluation, feature selection, filter approach, HDLSS data, wrapper approach.

The associate editor coordinating the review of this manuscript and approving it for publication was Jon Atli Benediktsson.

I. INTRODUCTION

High-dimensional, low-sample-size (HDLSS) datasets have a large number of features and a small number of samples, often containing complex nonlinear patterns. If the dataset contains s samples and the length of features is d, then this type of data exhibits the property $d \gg s$, indicating a very low ratio between s and d. During the era of big data, there has been a noticeable emergence of HDLSS data, finding utility across various applications such as disease diagnosis [1], biomarker discovery [2], drug development [3], natural language processing (NLP) [4], image and signal processing [5], environmental studies [6], and finance and stock market analysis [7]. The imbalance between high dimensionality and limited sample size in HDLSS data commonly presents significant challenges in data analysis and modeling. This results in increased storage costs, longer model building and execution times, decreased model interpretability, and it adversely affects the performance of the majority of learning algorithms [8]. In most instances, these challenges arise due to the existence of irrelevant or redundant features [9]. Hence, feature selection can be one of the most effective techniques to reduce the aforementioned problem. In addition, this is important because acquiring features is usually expensive and more features require more instances for learning to get good generalization capability of a classifier [10].

The primary objective of feature selection is to select a subset of relevant features, avoiding irrelevant, redundant, or noisy features from a large feature set, so that this reduced dimension of data is necessary and sufficient to describe the target concept [11]. Due to its ability to enhance the performance of classification algorithms, feature selection has been successfully applied in many real-world applications such as text classification, bio-informatics, image retrieval, intrusion detection, and so on [12]. However, finding the optimal feature subset is a challenging task, especially when dealing with a high number of features, as it requires extensive computational time to find the best solution. Fundamentally, it represents a combinatorial optimization problem of class NP-hard problem [13]. To address the challenge of feature subset selection, various approaches have been developed, which are categorized into three primary groups: wrapper, embedded, and filter methods. Each group employs different techniques for selecting the optimal features, and depending on the situation, each group has its own set of advantages and disadvantages. Wrapper approaches use a learning algorithm to evaluate the goodness of selected feature subsets, whereas filter approaches incorporate statistical and information theories to evaluate feature subsets independently of any classification algorithm. Embedded approaches select features during machine learning model building. It is observed that filter approaches are computationally efficient than wrapper approaches, but in terms of classification accuracy, wrappers are better than filters [14].

In the wrapper-based feature selection approach, a search strategy is incorporated to find the optimum subsets of features. Exhaustive search, greedy search, and metaheuristic are the popular options for this purpose [15]. Exhaustive search examines every conceivable subset of features to find the feature subset combination that yields the best performance. Although it ensures finding the best subset, for datasets with many features, it can be computationally expensive. On the other hand, greedy search often results in sub-optimal solutions since it only considers the best local option at each stage, without planning ahead or considering all options. Although greedy algorithms are computationally efficient, getting trapped in a local optimum can be a problem. Metaheuristics, however, have the capability of finding a near-optimal feature subset within a reasonable time frame, and they are popular approaches in literature for wrapper-based feature subset selection. Metaheuristics are problem-independent algorithms that guide subordinate heuristics with some intelligent strategies for exploring and exploiting the search space so that an efficient solution (optimal or near-optimal) can be found. They have recently been successfully applied to real-world and combinatorial optimization problems. Metaheuristics are broadly classified into trajectory-based and population-based approaches. Trajectory-based approaches use a single solution at a time to solve optimization problems, whereas population-based approaches consider a set of solutions to address optimization tasks. Some examples of trajectory-based metaheuristic approaches such as Simulated Annealing (SA) [16], and Tabu Search (TS) [17]. Genetic Algorithms (GA) [18], Particle Swarm Optimization (PSO) [19], and Differential Evolution (DE) [20] are some examples of popular population-based search.

Classification is an important data mining task that predicts categorical labels. Data classification is a two-step approach. The first step is the learning of a mapping or function built from training dataset, whereas in prediction phase map function performs on the feature represented data to predict the labels. These methods can be categorized broadly into linear classifiers, probabilistic, support vector machines, decision trees, and neural networks. It is observed that Support Vector Machine (SVM) is effective when the number of dimensions is greater than the number of samples [21].

Classifying HDLSS data presents a number of difficulties. The primary concern is the possibility of overfitting due to a large number of features and a small sample size. This creates challenges in improving generalization performance and avoiding overfitting. Additionally, there is the challenge of identifying relevant features from a vast feature set, as many features in these datasets are irrelevant or redundant [22]. It is highly likely that a classification algorithm cannot solely discover complex nonlinear patterns in data containing a large number of features. Therefore, a preprocessing step is usually required to deal with this type of data, such as performing feature selection or dimensionality reduction. Filter-based ranking feature selection can be used to select the top number of features, but the approach cannot handle redundant features properly [23].

Sometimes relatively low-ranked features can combine with high ranked features and contribute to better classification performance.

Another approach involves the feature subset selection from the original set using a wrapper-based search strategy. Searches such as sequential, exhaustive, greedy, or random search may not be effective as they are either computationally expensive or do not produce a globally optimal solution [24]. Therefore, for HDLSS datasets, a wrapper-based metaheuristic search might be a better option to find an optimal or at least a sub-optimal feature set at a reasonable computational cost. However, relying solely on a wrapper-based metaheuristic search may not be effective due to the presence of redundant and irrelevant features in feature sets, which can hinder convergence speed and the quality of feature subsets. Therefore, removing irrelevant and redundant features with filter approaches, followed by a wrapper-based search strategy, could be a better approach to address the feature selection problem.

In this work, we have developed a two-step approach to address feature subset selection for HDLSS data. Initially, an ensemble feature selection combining five filter approaches–Chi-square (χ^2), Gini index (GI), F-score, Mutual Information (MI), and Symmetric uncertainty (SU)– is developed to identify the candidate feature subset. The ensemble feature selection is employed to effectively filter out irrelevant and redundant features from the dataset. In the subsequent step, we incorporate Differential Evolution (DE)-based metaheuristic to identify the optimal feature subset. DE is chosen due to its effectiveness as an algorithm for finding optimum feature subsets [25]. Various improved variants of DE have been proposed to address the problem of feature selection [26]. However, the integration of an effective filter approach with wrapper-based DE for feature selection in HDLSS data has not been extensively explored. The primary aim of this study is to design a feature selection approach based on a filter-wrapper method to extract an appropriate subset of features for the classification of HDLSS data. To fulfill the aim the following objectives are considered:

- To Analyze how ensemble feature selection remove the unnecessary features and can enhance the performance of the HDLSS data classification task.
- To Investigate the effectiveness of combining the ensemble feature selection with a wrapper-based DE algorithm to select the best feature subsets that improve the classification accuracy of the classification model.

The main contributions of this paper are as follows:

- We propose a two-step feature subset selection strategy that combines ensemble feature selection with a wrapper-based search for HDLSS data. In the first step, five univariate filter-based feature selections are used to develop an ensemble feature selection for removing irrelevant features.
- In the second step, to attain the optimal feature subset, the efficient wrapper-based search is designed with a DE-based metaheuristic strategy and a fitness function that combines the weighted SVM classification error rate with the ratio of feature cardinality.
- The proposed approach is also capable of significantly reducing the number of features while maintaining excellent classification accuracy compared with state-of-the-art methods.

Section I presents the introduction of the paper. The remainder of the paper is organized as follows: the related works that discuss approaches to handle HDLSS data are outlined in Section II. The proposed approach is described in detail in Section III, where ensemble feature selection is presented after wrapper-based approach. Section IV presents the experimental setup for the study. The findings and discussion of the study are presented in Section V, and the conclusion and potential directions for further research are outlined in Section VI.

II. RELATED WORKS

In recent years, many significant research efforts have been produced to study the HDLSS data classification using different feature selection techniques. Some of the key previous research methods have been summarized below.

To address high-dimensional datasets, Ben Brahim and Limam [27] introduced a filter ensemble feature selection method based on the assessment of feature selectors' reliability. In this approach, both homogeneous and heterogeneous ensembles are employed to derive diverse feature subsets. The ensemble outputs of the selectors are subsequently combined using a robust aggregation technique that takes into account classification performance and reliability assessment. Experiments revealed that this method frequently enhances the stability and classification performance for HDLSS data sets, or at least maintains the baseline results when they are particularly high. In HDLSS data, Tsai and Sung [28] developed ensemble feature selection. The results demonstrate that, in terms of classification accuracy, ensemble feature selection outperforms single feature selection. However, there are no appreciable performance differences between the top three parallel and serial combinations and the best baseline method (GA).

To work directly on HDLSS data sets without dimensionality reduction, Shen et al. [29] proposed a novel linear binary classifier called the no-separated data maximum dispersion classifier. However, the method can be challenging to implement and comes with a high computational cost.

Liu et al. [30] introduced a Deep Neural Network (DNN) approach designed for HDLSS data. Their proposed solution is an end-to-end DNN model named Deep Neural Pursuit (DNP). By utilizing the multiple dropout technique, DNP can effectively learn from a limited number of samples and remains stable for feature selection. Empirical evidence supports its strong performance in both classification and feature selection tasks. In a different approach, Chowdhury et al. [31] proposed the use of Recurrent Neural Networks (RNN) for the purpose of feature selection. The proposed framework was employed for feature selection in microarray data used for cell classification. Four distinct feature selection models, each with a different architecture of recurrent neural networks, were introduced within this framework. These architectures encompass the Gated Recurrent Unit (GRU), Long-Short-Term Memory (LSTM), standard RNN, and the Bi-directional LSTM. The research findings demonstrate that this deep learning approach is well suited for classifying HDLSS data. Research output reveals that the proposed deep learning approach is suitable for HDLSS data classification.

In order to select informative features, a four-stage feature selection strategy has been proposed in [32] to solve the problem of classification of high-dimensional data. To obtain the final subsets of features, the proposed method first chooses candidate features using a variety of filtering techniques based on various metrics. Next, it applies semi-wrapper, union, and voting stages, in that order. A variety of statistical learning and data mining techniques has been used to test the effectiveness of the chosen features. Bommert et al. [33] conducted an empirical investigation into the operation of various filter methods for feature selection over HDLSS data. These filter methods were combined with a classification method, and 22 of them were evaluated in terms of runtime and precision using 16 high-dimensional datasets. The empirical study revealed that no single class of filter methods consistently outperforms all others.

Jain et al. [34] developed a two-phase hybrid model for cancer classification where Correlation-based Feature Selection (CFS) was integrated with improved Binary Particle Swarm Optimization (iBPSO). Experiment was performed with DNA microarray technology, and the proposed model exhibited better results in terms of classification accuracy and the number of genes selected in many cases. A method based on Artificial Bee Colony (ABC) and gradient boosting decision trees was proposed by Rao et al. [35] with the aim of addressing informative quality of the chosen features in HDLSS data. To improve the effectiveness of wrapper-based feature selection techniques, Thaher et al. [36] proposed a binary variant of the Harris Hawks Optimizer (HHO). Numerous tests and comparisons show that HHO is more stable when handling HDLSS type datasets. In another study, a comprehensive comparative analysis was performed using three well-known nature-inspired metaheuristic algorithms. Binary Particle Swarm Optimization (BPSO), GA and Cuckoo Search (CS) algorithm [37]. These algorithms were used for gene selection and classification in twelve high-dimensional cancer datasets. The methodology was carried out through the utilization of a three-phase hybrid approach to get better efficiency. Ghosh et al. [38] proposed a wrapper-filter combination of Ant Colony Optimization (ACO), where they introduce the evaluation of the subset using a filter method instead of using a wrapper method to reduce computational complexity. A memory to keep the best ants and feature dimension-dependent pheromone update have also been used to perform feature selection in a multi-objective manner. More recent relevant studies that have been using HDLSS datasets for classification problems include [39], [40], [41], [42], [43], and [44].

In the aforementioned study, it is discerned that HDLSS data often present class imbalance issues. Many of the above studies do not specify how they dealt with these imbalanced datasets to avoid probable overfitting of data. The crafty use of both the filter and wrapper approaches is imperative to find the optimum feature subset to improve the HDLSS classification problem, a challenge faced by many existing approaches. Moreover, properly balancing the exploration and exploitation of wrapper-based search also plays an important role in discovering the best feature subsets. Considering these challenges, our paper focuses on finding the optimum feature subset that can improve the model's performance, considering a combination of wrapper and filter-based approaches.

III. PROPOSED APPROACH

In this section, we first present an overview of the proposed filter-wrapper feature subset selection approach, followed by a detailed description. Figure 1 illustrates the overall methodology of the proposed approach. First, HDLSS datasets are collected, and several data preprocessing tasks are performed, including data cleaning, missing value imputation, data balancing, and data normalization. The data is also partitioned into training and testing sets. The next step involves selecting the candidate feature set using an ensemble feature selection approach. In this ensemble approach, five univariate filter-based methods–Chi-square (χ^2), Gini index (GI), F-score, Mutual Information (MI), and Symmetric uncertainty (SU)–are used to generate the top K features. The reduced datasets, based on the candidate feature set, are employed in the next phase to select the optimal feature subset. A wrapper-based approach is employed, wherein the DE metaheuristic serves as the search procedure, and a weighted sum of the SVM classifier error rate and the ratio of feature cardinality is used as the fitness measure for each feature subset during the optimization phase. Finally, datasets containing optimal feature subsets are evaluated using three classifiers: SVM, K-Nearest Neighbors (KNN), and Logistic Regression (LR).

A. DATASET COLLECTION AND PREPROCESSING

In this study, we collected thirteen HDLSS datasets. The datasets ALL-AML-3, ALL-AML-4, CNS, Colon, Lymphoma, MLL, and SRBCT are sourced from the website https://csse.szu.edu.cn/staff/zhuzx/datasets.html. Additionally, the remaining datasets are obtained from https://file.biolab.si/biolab/supp/bi-cancer/projections/. Table 1 illustrates the datasets chosen for this study, providing information on their respective names, the number of features, the number of samples, and the number of classes. It is clearly observed that number of features is much higher than number of instances.

Many of the datasets are not appropriate to use for machine learning model and require data prepossessing.

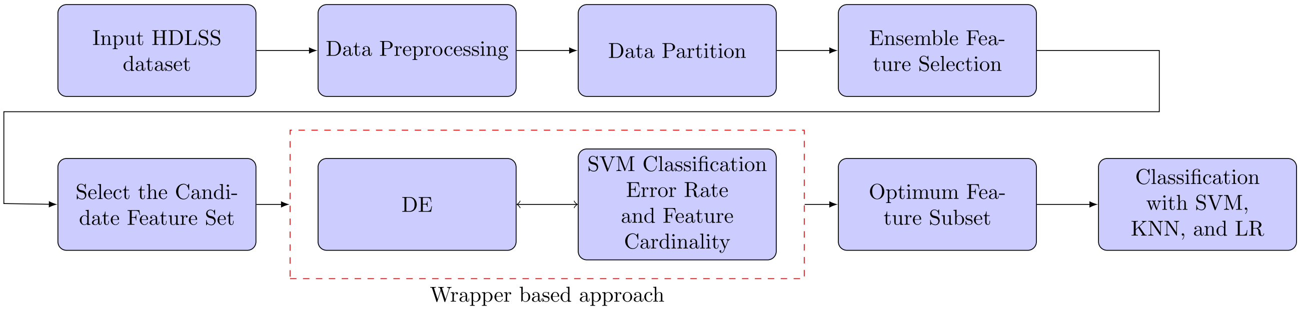

Input HDLSS dataset
Data Preprocessing
Data Partition
Ensemble Feature Selection
Select the Candidate Feature Set
DE
SVM Classification Error Rate and Feature Cardinality
Optimum Feature Subset
Classification with SVM, KNN, and LR
Wrapper based approach

FIGURE 1. Block diagram of the proposed framework.

TABLE 1. Dataset.

Datasets	No. of features	No. of Samples	No. of Classes
ALL-AML-3	7129	72	3
ALL-AML-4	7129	72	4
Braintumor	7129	40	5
Glioblastoma	12625	50	4
CNS	7129	72	2
Colon	2000	60	2
DLBCL	7070	77	2
Leukemia	5147	72	2
Lung	12600	203	5
Lymphoma	4026	62	2
MLL	12582	72	3
Prostate	12533	102	3
SRBCT	2308	83	4

We resolve any constant features or quasi-constant features using variance threshold methods. We eliminated features with more than 99% similar values for quasi-constant features. It is worth noting that some datasets contain missing values, which are addressed through the Median Imputation method, replacing missing values with the median value of the respective feature. Before proceeding with feature reduction, normalization is applied to each set. We apply the Min-Max scaling method to normalize the data within a range of 0 to 1 using the formula:

$$X_i^{nor} = \frac{X_i - Xmin}{X_{max} - X_{min}} \tag{1}$$

In this equation, X_i^{nor} denotes the normalized value of the i^{th} element of vector $X = [X_1, \ldots, X_n]$, while X_{min} and X_{max} represent the minimum and maximum values of the vector respectively.

As the data are not balanced, we apply random oversampling to address class imbalance in a dataset. In this process, we artificially increase the number of instances for each minority class until the class distribution is balanced, ensuring that all classes have an equal number of instances. Table 2 displays the class distribution of the datasets before and after balancing. The '-' symbol indicates that the corresponding datasets do not have this class label. As shown in the table, before balancing, the ALL-AML-3 dataset has three classes (Class 0, Class 1, Class 2) with instances numbering 25, 38, and 9, respectively. After balancing, all three classes have an equal number of instances, which is 38. Similarly, the same process is performed for other datasets.

B. ENSEMBLE FEATURE SELECTION

After the basic preprocessing, HDLSS datasets still contain a large number of features. In the first stage of our proposed approach, we employ an ensemble feature selection strategy to select the initial candidate features. Five univariate filter approaches have been utilized, including χ^2, GI, F-score, MI, and SU filter rankings for the ensemble feature selection task. In the feature selection using univariate filter ranking, individual features are evaluated independently and then ordered based on their relevance to the target variable. Within this ensemble approach, each filter ranks the features based on their importance and sorts them according to their rank value. It is evident that each method might rank the same feature with a different value. Hence, we combine all ranked lists into a single final decision. The process is straightforward: for each feature, all ranking values calculated using filter approaches are selected, and then the median value among them is considered the final ranking of that feature. Median value is used because the median is less influenced by any extreme value. In this manner, the final ranking values for all features are calculated. In other words, given the full feature set $F = \{f_1, f_2, \ldots, f_d\}$, where f_i is i^{th} feature of d dimensional feature set. Suppose there are M feature selectors such that $R = \{R_1, R_2, \ldots, R_M\}$, where R_i is the i^{th} feature selector (i.e., ranker). If rank value of f_i^{th} feature for R_j^{th} ranker is $w_{f_i}^{R_j}$, then the final rank of f_i^{th} feature in ensemble feature selection will be $Median\{w_{f_i}^{R_1}, w_{f_i}^{R_2}, \ldots w_{f_i}^{R_M}\}$. The features are then sorted according to their ranking value in descending order. Finally, the top k features are selected as candidate feature sets. Figure 2 depicts the ensemble feature selection that has been proposed.

At this stage, we also examine the impact of the ensemble feature selection on model performance when SVM classifier is used for evaluation. This reduced feature set serves as an input for the next step in obtaining the optimal feature subset.

It should be noted that we have utilized the aforementioned five filter methods due to their popularity in feature selection.

TABLE 2. Class Distribution of the datasets before and after balancing.

Datasets	Before Balancing					After Balancing				
	Number of Instances in Target Class					Number of Instances in Target Class				
	Class 0	Class 1	Class 2	Class 3	Class 4	Class 0	Class 1	Class 2	Class 3	Class 4
ALL-AML-3	25	38	9	-	-	38	38	38	-	-
ALL-AML-4	38	21	4	9	-	38	38	38	38	-
Braintumor	4	6	10	10	10	10	10	10	10	10
CNS	21	39	-	-	-	39	39	-	-	-
Colon	40	22	-	-	-	40	40	-	-	-
DLBCL	58	19	-	-	-	58	58	-	-	-
Glioblastoma	14	7	14	15	-	15	15	15	15	-
Leukemia	47	25	-	-	-	47	47	-	-	-
Lung	139	20	17	6	21	139	139	139	139	139
Lymphoma	58	19	-	-	-	58	58	-	-	-
MLL	24	28	20	-	-	28	28	28	-	-
Prostate	50	52	-	-	-	52	52	-	-	-
SRBCT	8	23	12	20	-	23	23	23	23	-

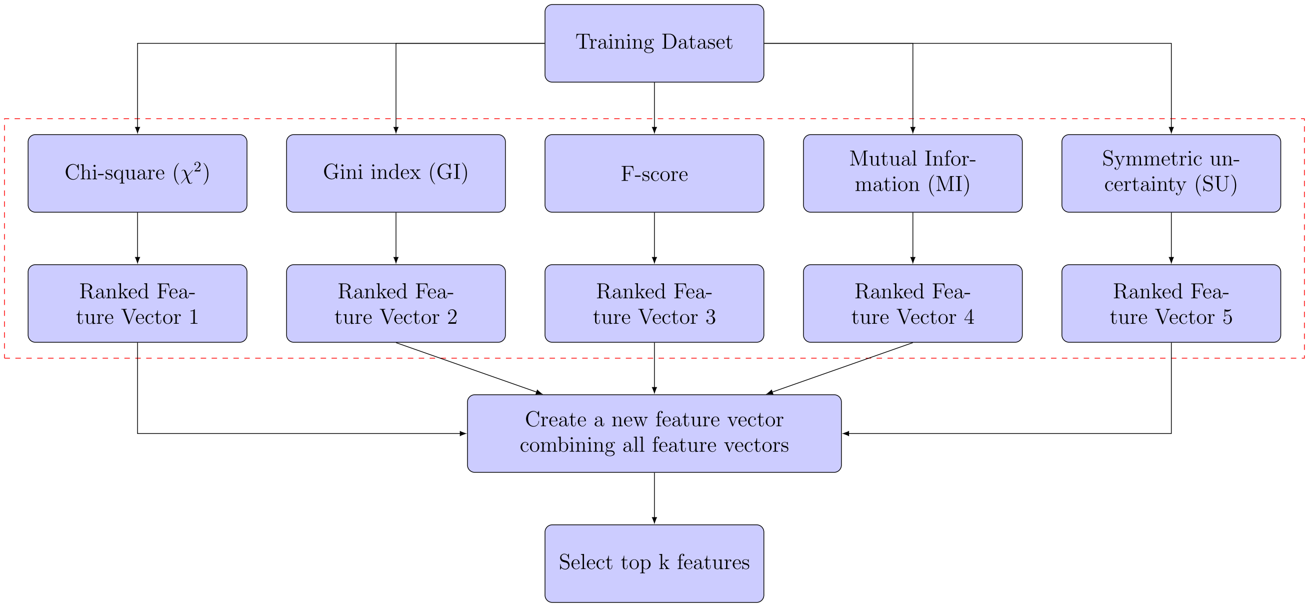


FIGURE 2. Proposed ensemble feature selection.

It is also observed that relying solely on a single ranking approach may not always guarantee the most appropriate rankings [45]. Therefore, the aggregation of these filter algorithms might produce superior performance in the HDLSS classification task. For the reader's convenience, we have provided a brief description of the filter approaches we adopted in this study.

1) CHI-SQUARE (χ^2)

Chi-square (χ^2) [46] is a statistical method for feature selection where it evaluates each feature x by measuring their χ^2 statistic with respect to the label variable y. The higher value indicates that the feature is more important. Mathematically, it is represented as:

$$\chi^2(x) = \sum_{i=1}^{r}\sum_{j=1}^{c}\frac{(O_{ij} - E_{ij})^2}{E_{ij}} \tag{2}$$

In this equation, r represents various feature values, while c denotes the class labels. The variable O_{ij} symbolizes the observed frequency. On the other hand, E_{ij} represents the expected frequency, signifying the anticipated number of observations for a particular class if there was no correlation between the feature and the target.

2) GINI INDEX (GI)

Gini index (GI) [47] is used in the selection of features to determine which features work best to divide a data set into subsets that are as similar to each other as possible with respect to the target classes. A smaller Gini Index value indicates greater relevance of the feature. The Gini index for a node n is computed as follows:

$$GiniIndex(n) = 1 - \sum_{i=1}^{C}[p(i|n)]^2 \tag{3}$$

where C represents the total number of classes, and $p(i|n)$ denotes the probability of an element being classified as class i in node n.

3) F-SCORE

F-score [48] is a simple univariate feature selection approach that assesses the discrimination of two sets of real numbers. Let the training vectors be x_k, $k = \{1, 2, \ldots, m\}$, number of positive and negative instances are n_+ and n_- respectively, the F-score is defined as:

$$F_i = \frac{(\bar{x}_i^{(+)} - \bar{x}_i)^2 + (\bar{x}_i^{(-)} - \bar{x}_i)^2}{\frac{1}{n_+-1}\sum_{k=1}^{n_+}(x_{k,i}^{(+)} - \bar{x}_i^{(+)})^2 + \frac{1}{n_--1}\sum_{k=1}^{n_-}(x_{k,i}^{(-)} - \bar{x}_i^{(-)})^2} \tag{4}$$

where $\bar{x}_i$ is the mean of the i^{th} feature across the entire dataset, $\bar{x}_i^{(+)}$ is the mean of the i^{th} feature for positive instances, and $\bar{x}_i^{(-)}$ is the mean of the i^{th} feature for negative instances. $x_{k,i}^{(+)}$ represents the i^{th} feature of the k^{th} instance in the positive dataset, while $x_{k,i}^{(-)}$ represents the i^{th} feature of the k^{th} instance in the negative dataset. A higher F-score indicates a more discriminatory feature.

4) MUTUAL INFORMATION (MI)

Mutual Information (MI) [49] is used to rank features based on the relationship between the feature X and the target class Y. The mathematical equation for Mutual Information between variables X and Y is represented as:

$$I(X;Y) = \sum_{y\in Y}\sum_{x\in X} p(x,y)\log\frac{p(x,y)}{p(x)p(y)} \tag{5}$$

where $I(X;Y)$ indicates the mutual information between variables X and Y, $p(x,y)$ refers to the joint probability distribution function of X and Y, and $p(x)$ and $p(y)$ are the marginal probability distribution functions of X and Y, respectively.

5) SYMMETRIC UNCERTAINTY (SU)

Symmetric Uncertainty (SU) [50] is a filter-based feature selection method that quantifies the relevance score between a feature and a class label. Essentially, SU functions as a correlation measure based on normalized mutual information [51]. The range of the metric spans from 0 to 1, with a higher value indicating a stronger association between the variables. Mathematically, SU can be represented as follows:

$$SU(X;Y) = \frac{2 \times I(X;Y)}{H(X) + H(Y)} \tag{6}$$

where $I(X;Y)$ is the mutual information between X and Y, $H(X)$ is the entropy of X, and $H(Y)$ is the entropy of Y. The entropy $H(X)$ of variable X is defined as:

$$H(X) = -\sum_i P(x_i)\cdot\log_2(P(x_i)) \tag{7}$$

where x_i represents the possible values of the random variable X, and $P(x_i)$ represents the probability of each corresponding value. A similar equation can be written for $H(Y)$.

C. WRAPPER-BASED FEATURE SELECTION

In the second step, we consider population based meta-heurstic Differential Evolution (DE) to find the optimal feature subset. Here, the main steps involving the encoding of a feature subset, the definition of the fitness function, the DE and its application in achieving the optimal feature subset have been presented in detail.

1) ENCODING OF A FEATURE SUBSET

To obtain an optimal subset of features during DE, the first step is to initialize a set of feature subset vectors. Then these vectors are iteratively updated and evaluated until a final solution is achieved. To efficiently execute this task, the feature subset needs to be encoded in an operational form suitable for the algorithm. The commonly used method is binary encoding, which is considered in this work. In this encoding, each feature can be regarded as a binary decision variable, denoted as f_i, such that:

$$f_i = \begin{cases} 0 & \text{if } f_i \text{ is not selected} \\ 1 & \text{if } f_i \text{ is selected} \end{cases} \tag{8}$$

For example, consider the feature set $F_T = \{f_1, f_2, f_3, f_4, f_5, f_6\}$. At a particular point, a feature subset $F_s = \{f_1, f_2, f_4\}$ is represented as the binary vector $F_{sb} = [110100]$, indicating that features f_3, f_5, and f_6 are not included in the subset F_{sb}. This encoding allows us to easily represent any feature subset vector as a binary feature subset vector.

2) FITNESS FUNCTION FOR EVALUATION

To evaluate the feature subset during the search, we consider two different objectives: classification error rate and the number of selected features. It is often challenging to simultaneously minimize the classification error rate and the number of features, as they frequently conflict, necessitating an optimal decision to balance between them. To gauge the quality of the feature subset during the search process, we employ a fitness function or objective function. This function represents a linear combination of the error rate and the ratio of feature cardinality. The ratio of feature cardinality is the proportion of selected features in the subset compared to the total number of features available in a dataset. Minimizing this objective function indicates that a feature vector with lower classification error and fewer selected features performs better, as shown in Equation 9:

$$fitness = \alpha * Error_{rate} + \beta * \frac{F_S}{F_T} \tag{9}$$

where α is a value in the range [0, 1], and $\alpha + \beta = 1$. The $Error_{rate}$ represents the classification error rate, F_S denotes the number of selected features, and F_T signifies the total number of features. The variables α and β are weighted values that indicate the importance of their respective components. It is worth noting that, in our experiment, we chose $\alpha = 0.90$, which places greater weight on the classification error rate.

3) PROPOSED DIFFERENTIAL EVALUATION

DE is a population-based metaheuristic algorithm developed to solve discrete and continuous optimization problems [52]. DE is a type of evolutionary algorithm that generates new candidate solutions through a multi-parent reproduction strategy. Unlike GA, in DE, crossover is applied after mutation, and it is a routine operation used to generate each offspring. In DE, a population of possible solutions is randomly generated, with each vector indicating a possible solution of a given problem. Mutation operation is performed in current population to generate new candidate solutions. Each mutant vector corresponding to each population member is created using three random solutions of the current population based on a mutation operation (a base vector and two difference vectors). The next crossover is performed in which elements of the mutant vector are crossed over with those of a target vector from the population to form a trial vector. A crossover rate parameter is defined and determines the probability that each element is inherited from either the mutant or the target vector. In the selection process, if a trail vector is better than that of the target vector, it replaces the target vector in the next generation. Otherwise, the target vector remains in the population. It is noted that the fitness function associate with the optimization problem is used to find the goodness of fit of a solution vector. The algorithm repeats the mutation, crossover, and selection specified number of generations or until a termination condition is met. The time complexity of DE is given by $O(I \times N \times D + O(f(\cdot)))$, where $f(\cdot)$ represents the complexity of computing the fitness function, N stands for the size of the population, I denotes the number of iterations, and D signifies the number of dimensions [53], [54]. Figure 3 illustrates the flowchart of the DE process.

4) DE FOR FEATURE SELECTION

To perform feature subset selection using DE, a set of randomly generated feature vectors is considered as the initial solution, where the individual feature subset vector is a binary string. Assume the population of n feature vectors with d dimension is represented as $X(t) = [X_1(t), X_2(t), \dots, X_n(t)]$. Then the i^{th} feature vector in the current population is as follows:

$$X_i(t) = [x_i^1(t), X_i^2(t), \dots, X_i^d(t)] \tag{10}$$

where $x_i^j(t)$ is the j^{th} feature in i^{th} feature vector at iteration t. At this initial stage $t = 0$, each $x_i^j(t)$ is assigned a uniform random number from $\{0, 1\}$ which can be written as follows.

$$x_i^j(t) = LB + Rand(0, 1)(UB - LB) \tag{11}$$

where LB (Lower Bound) is 0 and UB (Upper Bound) is 1, and $Rand(0, 1)$ denotes a random selection between the discrete values 0 and 1, each with an equal probability of being chosen.

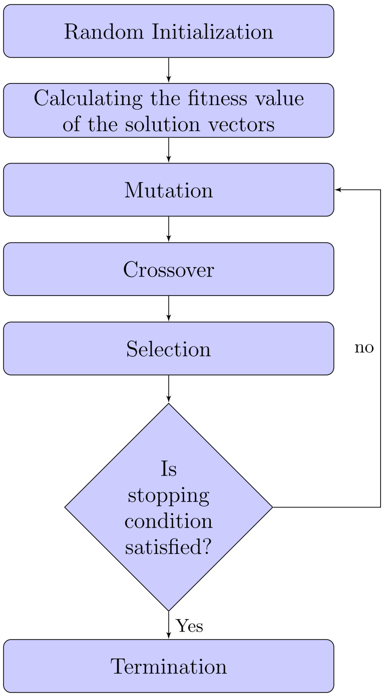

Random Initialization
Calculating the fitness value of the solution vectors
Mutation
Crossover
Selection
no
Is stopping condition satisfied?
Yes
Termination

FIGURE 3. Block diagram of the DE.

To perform the mutation, we have used a simple mutation operation named DE/rand/1. For the population with N individuals, the i^{th} mutation vector $Y_i(t) = [y_i^1(t), y_i^2(t), \dots, y_i^d(t)]$ is generated corresponding to target vector $X_i(t)$ as follows:

$$Y_i(t) = X_{r1}(t) + F \cdot (X_{r2}(t) - X_{r3}(t)) \tag{12}$$

where t is the t-th generation. The three vectors, denoted $X_{r1}(t)$, $X_{r2}(t)$, and $X_{r3}(t)$, are randomly selected from the population, with $r1 \neq r2 \neq r3$. F is a scaling factor used to control the mutation process and has a value in the range between 0 and 1.

After mutation operation, it is quite common that the value of the feature vector is numerical and we need to convert it to binary. This task is performed using the following equation.

$$y_i^j(t) = \begin{cases} y_i^j(t) = 1 & \text{if } y_i^j(t) > Th \\ y_i^j(t) = 0 & \text{if } y_i^j(t) \le Th \end{cases} \tag{13}$$

where Th indicates a threshold value.

Crossover is also performed where a trial vector $Z_i(t) = [z_i^1(t), z_i^2(t), \dots, z_i^d(t)]$ is produced using a mutation vector $Y_i(t)$ and a population target vector $X_i(t)$. We consider the following crossover function.

$$z_i^j(t) = \begin{cases} y_i^j(t) & \text{if } rand(0, 1) \le C_r \text{ or } j = j_{rand} \\ x_i^j(t) & \text{otherwise} \end{cases} \tag{14}$$

where $j_{rand} \in \{1, \ldots, d\}$ is randomly chosen integer, and C_r is the crossover rate, such that $C_r \in [0, 1]$. We choose it to be 0.5.

In order to select the population for the next generation, the fitness function is used to compare the trial vector $Z_i(t)$ with target vector $X_i(t)$. The trial vector is selected only when its fitness is better (less than) that target vector. Therefore, the feature vector $X_i(t+1)$ for next iteration is produces as follows:

$$X_i(t+1) = \begin{cases} Z_i(t) & \text{if } \mathit{fitness}(Z_i(t)) < \mathit{fitness}(X_i(t)) \\ X_i(t) & \text{otherwise} \end{cases} \tag{15}$$

The pseudo-code for the proposed DE-based feature subset selection algorithm for HDLSS datasets is presented in Algorithm 1. The primary objective is to iteratively evolve a population of feature vectors, seeking the optimal subset of features that minimize a predefined fitness function. The algorithm initiates by initializing the population of feature vectors, followed by the evaluation of each vector's fitness using a specified fitness function. The variable f_{best} is then initialized to the feature subset with the minimum fitness among the initial population. Over a specified number of iterations, the algorithm proceeds to generate mutation vector for each feature vector, convert this vector to binary representation, and create trial vector through a crossover function. Subsequently, the fitness of each trial vector is evaluated, and the population for the next iteration is selected based on a selection procedure. The algorithm continuously updates f_{best} to reflect the feature subset with the minimum fitness in the evolving population. The final result is the best feature subset f_{best} found after the specified number of iterations.

Algorithm 1 Feature Subset Selection Using DE-Based Approach

```
Input: HDLSS Training data with top $k$ features, Population
    size $n$, Dimension of each feature vector $d$, Maximum
    number of iterations $T_{\max}$, Current iteration $t$, Best
    feature subset $f_{\text{best}}$
Output: Best feature subset $f_{\text{best}}$
    Initialisation:
for Each feature vector of the population do
    Initialize feature vector by Eq. 11
    Evaluate the fitness of feature vector by Eq. 9
end for
    iteration Process
$f_{best}$ ← feature subset with the minimum fitness
for $t = 1$ to $T_{\max}$ do
    for Each feature vector of the population do
        Generate mutation vector by Eq. 12
        Generate binary transformation of each element of
            a mutation vector by Eq. 13
        Generate trial vector using crossover function by
            Eq. 14
        Evaluate the fitness of the trial vector by Eq. 9
        Select the population for the next iteration by Eq. 15
    end for
    $f_{\text{best}}$ ← feature subset with the minimum fitness so far
end for
return $f_{best}$
```

After the selection of the feature subset using a wrapper-based DE search, the training set of this particular feature subset from the HDLSS dataset is incorporated to construct the classification model. SVM, KNN, and LR are employed for this model-building task. Subsequently, the test dataset, which contains the same selected features, is used for the classification task.

IV. EXPERIMENTAL SETUP

We conducted our experiment on 13 HDLSS datasets, as shown in Table 1. Each dataset is divided into training and testing sets, with 80% of the data allocated to the training set and 20% to the testing set, based on 5-fold cross-validation. Feature selection is performed on the training sets. Additionally, during the wrapper-based search for measuring classification error rate, we employed 5-fold cross-validation. K-fold cross-validation is chosen to mitigate the risk of overfitting. Regarding performance measures, both Area Under the Curve (AUC) and Classification Accuracy (ACC) have been utilized. AUC is used because it proves to be a valuable metric in scenarios involving imbalanced class distributions and varying threshold requirements. AUC and ACC have been measured in percentage and their equations are shown below.

$$ACC\% = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \times 100 \tag{16}$$

and

$$AUC\% = \frac{\text{TPR} + \text{FPR}}{2} \times 100 \tag{17}$$

where TP is True Positive, FP is False Positive, FN is False Negative, TN is True Negative,

$$\text{TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \quad \text{and} \quad \text{FPR} = \frac{\text{FP}}{\text{FP} + \text{TN}}.$$

In our approach, we employed the SVM classifier with a linear kernel function for both determining the classification error rate in the wrapper-based search process and measuring the performance of the final classification model. In addition, we employed two other classifiers, LR and KNN, to measure the performance of the final models. For the LR classifier, we applied L2 regularization with a constant of 1. For the KNN classifier, we set the number of neighbors to 5. We selected the top 110 features as candidate features for ensemble feature selection. This choice was informed by preliminary experiments where feature number variations ranging from 5 to 140 with an interval of 15 features were tested. The results indicated that the top 110 features yielded satisfactory ACC for most datasets. During the DE-driven wrapper-based search, we set the population size to 25 and the maximum number of iterations to 200. The crossover rate (C_r), scaling factor (F), and threshold value (Th) of DE were all set to 0.5. These parameter choices were made through trial and error. The proposed approach was executed 20 times with a random seed on a Windows 11 64-bit operating system, running on an Intel Core i5-1035G1 CPU @ 1.00GHz with 8 GB of RAM. The implementation of the proposed method is in Python 3.5, utilizing various Python libraries, including Numpy, SciPy, Matplotlib, and Scikit-learn. Table 3 summarizes the different parameters set for this experiment.

TABLE 3. Experimental parameters.

Parameter	Values
Number of Runs	20
Population Size	25
Number of Iterations	200
α of Fitness Function	0.9
β of Fitness Function	0.1
k Value for Ensemble Feature Selection	100
Crossover Rate (C_r) for DE	0.5
Scaling Factor (F) for DE	0.5
Threshold Value (Th) for DE	0.5

V. RESULT AND DISCUSSION

In our first experiment, we compare the proposed ensemble feature selection with five other filter-based feature selection approaches (χ^2, GI, F-score, MI, and SU), as well as the approach without feature selection. Table 4 presents the ACC obtained for thirteen datasets using both feature selection and the approach without feature selection. To find the ACC, we used an SVM classifier. In the feature selection approaches, the top $k = 110$ features are selected for each dataset. The value of $k = 110$ is determined based on preliminary experiments. It is observed that, in general, the ACC for almost all datasets does not significantly improve beyond this value. The highest accuracy for each dataset bolded. It is also noted that the underlined value indicates the best results among the feature selection approaches. The results indicate that feature selection can significantly enhance ACC in the majority of HDLSS datasets. Without feature selection, the highest results are achieved for only two datasets, specifically the *colon* dataset, where the performance without feature selection surpasses any feature selection algorithm. Among the various feature selection methods, ensemble feature selection demonstrates the best performance, yielding the highest ACC for twelve datasets (excluding *ALL-AML-3*). The next best-performing feature selection method is GI, providing the best classification for 8 cases. Both χ^2 and F-score yield the best results for 7 cases, followed by SU and MI for 6 and 5 cases, respectively. The robustness and effective accuracy observed across almost all datasets with ensemble feature selection suggest that this approach can effectively eliminate irrelevant features compared to other feature selection methods.

Figure 4 displays the ACC performance of the ensemble feature selection approach with the number of selected features. We vary the number of top-selected features from 5 to 140 with an interval 15. It is observed that when we select the top 5 features, the ACC for most datasets is very low. With the increase in the number of features, ACC improves for some datasets, but for the majority of datasets, it either fluctuates or remains stable. When the number of top features is set to around 100 or more, it is observed that, for most datasets, the accuracy values are stable or show slight improvement. Based on this analysis, we have concluded that selecting top 110 features for ensemble feature selection generally yields better performance across the majority of the datasets.

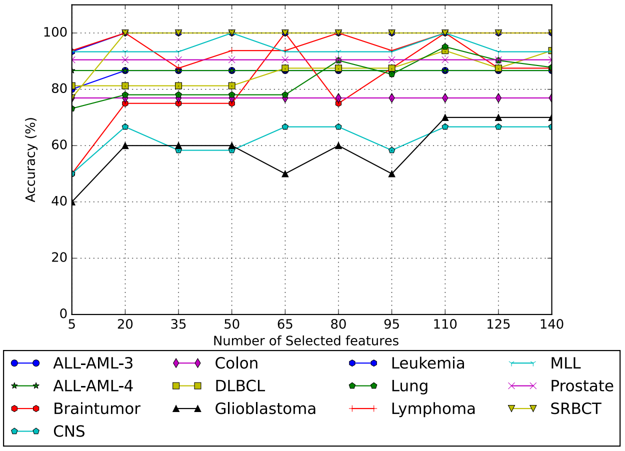


FIGURE 4. ACC with varying numbers of top-selected features when only ensemble feature selection is used.

Table 5 illustrates the performance of the proposed ensemble with a wrapper-based feature selection approach in terms of selecting the feature subset for HDLSS datasets. It presents the minimum, maximum, average, and standard deviation (SD) of the number of selected features. The average number of selected features for each dataset indicates that the proposed approach can significantly reduce the number of features. The relatively low SD for each dataset also indicates less variability or spread in the number of selected features. On average, the approach reduces the number of features within the range of 10.50 to 1.95, with seven cases (*ALL-AML-3*, *DLBCL*, *Leukemia*, *Lymphoma*, *MLL*, *Prostate*, and *SRBCT*) having an average selected number below 5.

To meticulously assess the effectiveness of our proposed ensemble method with a wrapper-based feature subset selection approach in classifying HDLSS datasets, we incorporated SVM, KNN, and LR classifiers to construct models for each dataset using the selected features. The ACC and AUC scores for SVM, KNN, and LR for each dataset are displayed in Table 6. The corresponding SD is also provided alongside each ACC and AUC score. Upon comparing ACC with AUC, it is observed that for all datasets and all

TABLE 4. ACC scores of SVM for datasets using ensemble feature selection, five filter-based feature selection approaches, and without feature selection.

Dataset	Without Feature Selection	Feature Selection					
		χ^2	GI	F-score	MI	SU	Ensemble feature selction
ALL-AML-3	66.67	86.67	**93.33**	86.67	86.67	86.67	86.67
ALL-AML-4	73.33	**86.67**	**86.67**	**86.67**	**86.67**	**86.67**	**86.67**
Braintumor	75.00	**100.00**	25.00	62.50	75.00	87.50	**100.00**
CNS	50.00	**66.67**	58.33	58.33	58.33	58.33	**66.67**
Colon	**84.62**	76.92	76.92	76.92	76.92	69.23	76.92
DLBCL	75.00	87.50	**93.75**	**93.75**	87.50	**93.75**	**93.75**
Glioblastoma	40.00	**70.00**	60.00	50.00	50.00	**70.00**	**70.00**
Leukemia	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**
Lung	85.37	95.12	90.24	78.05	90.24	95.12	**97.56**
Lymphoma	75.00	93.75	**100.00**	**100.00**	87.50	93.75	**100.00**
MLL	86.67	93.33	93.33	100.00	93.33	93.33	**100.00**
Prostate	76.19	85.71	**90.48**	**90.48**	**90.48**	**90.48**	**90.48**
SRBCT	92.31	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**

TABLE 5. Number of selected features using proposed ensemble with wrapper based approach.

Dataset	Number of Selected features			
	Minimum	Maximum	Average	SD
ALL-AML-3	2.00	6.00	3.90	1.26
ALL-AML-4	4.00	8.00	5.85	1.19
Braintumor	5.00	14.00	8.40	2.69
CNS	6.00	19.00	10.50	3.14
Colon	2.00	11.00	5.70	2.47
DLBCL	2.00	6.00	4.00	1.05
Glioblastoma	4.00	16.00	5.90	3.13
Leukemia	1.00	4.00	2.55	0.74
Lung	6.00	15.00	8.55	2.01
Lymphoma	1.00	4.00	1.95	0.67
MLL	2.00	6.00	3.35	1.11
Prostate	2.00	8.00	4.45	1.72
SRBCT	2.00	4.00	3.00	0.63

classifiers, the AUC is either slightly superior to ACC or equivalent. This suggests that the classification models have superior discriminatory capabilities. Generally, all classifiers achieve over 98% in average AUC and ACC scores when the datasets with selected features are used for classification. Among the three classifiers, it is also observed that the SVM classifier performs the best, achieving 100% ACC and AUC for 7 datasets, and nearly 100% in the remaining cases. The performance of KNN and LR is almost identical. Furthermore, in all instances, the SD values indicate that the ACC and AUC values are closer to the mean. It is noted that as SVM consistently outperforms other two classification models, we utilized the results obtained from the SVM classifier for subsequent comparisons with state-of-the-art approaches.

As mentioned earlier, DE-based metaheuristic has been employed as a search approach. The fitness value associated with the search guides the DE to converge towards the global optimum. In our approach, the objective is to minimize the fitness value, and convergence is deemed achieved when the fitness value remains unchanged. The convergence curve of the proposed algorithm on 13 benchmark datasets is depicted in Figure 5. In our approach, the number of iterations is set to 200. It is observed that the method converges to the optimal solution before completing 200 iterations, highlighting the fast convergence performance of the proposed approach.

Figure 6 summarizes the ACC obtained using three different approaches: without feature selection, ensemble feature selection, and the proposed ensemble with a wrapper-based approach (SVM). Without the feature selection approach, it is found that the ACC for HDLSS datasets cannot be improved; for the majority of datasets, the ACC is less than 90% for all datasets, with the exception of the *Leukemia* dataset. When ensemble feature selection is applied, it performs better than without the feature selection algorithm for all HDLSS datasets (apart from *Colon*). However, of the three approaches, the ensemble with the wrapper-based approach performed the best, producing accuracy around 100% for almost all datasets. Although the ensemble approach is producing competitive results in ACC for the *Braintumor*, *Leukemia*, *Lung*, *Lymphoma*, *MLL* and *SRBCT* datasets, it is observed that these results were obtained with an ensemble approach that selected 110 features, while the ensemble with the wrapper selected, on average, fewer than 10 features.

It is essential to assess the performance of the proposed approach with other relevant methods. Therefore, we consider nine state-of-the-art feature selection approaches in which these HDLSS datasets are used for classification tasks. To perform appropriate benchmarking, we compare our model with those approaches in which a metaheuristic-driven wrapper-based search is incorporated. Table 7 shows the description of the comparative methods in which the first column is the name of the approach, the second column is the description of the method, and finally the last column is the reference.

Table 8 shows the results of the comparison of the proposed approach with nine state-of-the-art methods. For comparison purposes, the average ACC and the average number of selected features (#*F*) are considered. It is noted that for each method only datasets with available results

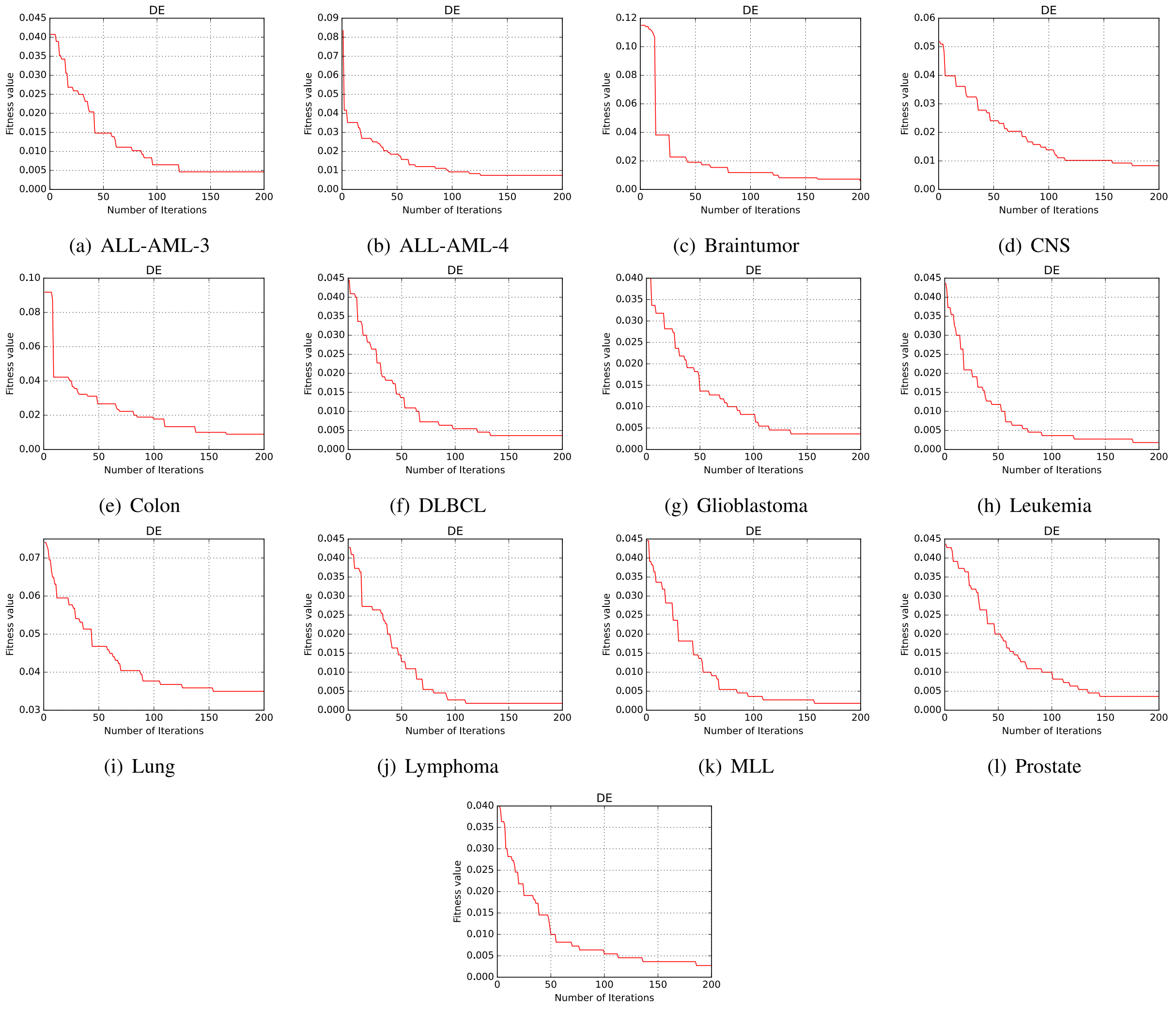
TABLE 6. ACC and AUC of KNN, SVM, and LR when the proposed ensemble with a wrapper-based approach is used for feature subset selection.

	KNN		SVM		LR	
Dataset	ACC	AUC	ACC	AUC	ACC	AUC
ALL-AML-3	98.29± 1.48	99.72±0.40	100.00±0.00	100.00±0.00	99.68±0.9	99.94±0.12
ALL-AML-4	98.63± 1.20	99.72±0.56	99.72±0.69	99.96±0.12	99.13±1.12	99.84±0.27
Braintumor	98.50± 2.00	99.17±0.76	99.17±0.88	99.22±1.42	98.67±2.81	98.83±1.40
CNS	98.33± 2.15	99.31±1.22	100.00±0.00	100.00±0.00	98.68±1.98	99.38±1.11
Colon	96.67± 1.79	97.19±1.68	98.95±1.71	99.1±1.54	98.75±2.01	99.65±0.88
DLBCL	99.14± 1.38	99.59±0.58	100.00±0.00	100.00±0.00	99.14±1.38	100±0.00
Glioblastoma	97.03± 2.23	98.15±1.14	99.67±1.05	99.67±1.05	93.33±3.51	96.31±1.82
Leukemia	98.62± 1.78	99.24±1.63	100.00±0.00	100.00±0.00	99.31±1.45	100±0.00
Lung	99.00± 0.57	99.77±0.21	99.33±0.75	99.96±0.07	98.47±0.95	99.86±0.11
Lymphoma	98.57± 3.09	99.92±0.26	100.00±0.00	100.00±0.00	99.71±0.90	100±0.00
MLL	98.85± 1.86	99.79±0.40	100.00±0.00	100.00±0.00	98.08±2.03	99.85±0.25
Prostate	97.81± 2.57	98.36±2.10	99.71±0.31	99.82±0.28	98.44±1.65	99.57±0.98
SRBCT	98.57± 1.84	99.71±0.75	100.00±0.00	100.00±0.00	97.14±1.51	98.66±1.42
Grand Average	**98.31± 1.97**	**99.20±1.29**	**99.73±0.73**	**99.83±0.69**	**98.35±2.39**	**99.38±1.30**

TABLE 7. Summary of the state-of-the-art approaches used for comparison.

Approach	Description	Reference
rMRMR-MGWO	The robust Minimum Redundancy Maximum Relevancy (rMRMR) method serves as a filter approach, while the modified Gray wolf optimizer algorithm (MGWO) method operates as a wrapper approach.	Alomari et al.(2021) [55]
QOMOJaya	Combining rankings from five filters helps to remove unnecessary genes, while the multi-objective Jaya algorithm is used for wrapper-based feature selection	Chaudhuri et al. (2022) [56]
MIM-mMFA	Mutual Information Maximization (MIM) operates as a filter approach, while the modified Moth Flame Algorithm (mMFA) is utilized for wrapper-based search	Dabba et al. (2021) [57]
mRMR-MBFA	The Max-Relevance and Min-Redundancy (mRMR) method is used for filter-based feature selection, while the improved Multi-layer Binary Firefly Algorithm (MBFA) is used for a wrapper-based feature selection approach	Xie et al. (2023) [58]
MBO-BLS	The Monarch Butterfly Optimization (MBO) is combined with the Broad Learning System (BLS) to identify optimum feature subsets. Before it, Relief is used to select candidate features.	Parhi et al· (2022) [59]
BCOOT-CSA	The framework uses the Coot Bird Optimization Algorithm (COOT) in conjunction with Simulated Annealing (SA) as a wrapper, and mRMR as a filter-based feature selection method.	Pashaei et al. (2023) [60]
CFC-FBBA	This approach uses the correlation-based feature clustering (CFC) algorithm as a filter-based feature selection method. It utilizes a modified version of the binary bat algorithm, named the fractional-order binary bat algorithm (FBBA), for the selection of wrapper-based features.	Esfandiari et al. (2022) [61]
BTLBOGSA	It employs mRMR for selecting relevant features and integrates the teaching-learning-based algorithm (TLBO) with the gravitational search algorithm (GSA) for wrapper-based feature selection.	Shukla et al. (2020) [62]
MFI-RFPA	A combination of three filters removes the majority of irrelevant and noisy features. In the subsequent stage, hybrid model of the multi-filter integration and recursive flower pollination algorithm (MFI-RFPA)is used to identify the optimal feature subset.	Li et al. (2023) [63]

are reported, while the symbol $'-'$ indicates that results are not available in the corresponding research papers. In addition, the best results are highlighted in bold. The *Glioblastoma* dataset is omitted from the table as none of these nine studies utilize it. In summary, our proposed method generally achieves competitive ACC while selecting



(a) ALL-AML-3 (b) ALL-AML-4 (c) Braintumor (d) CNS

(e) Colon (f) DLBCL (g) Glioblastoma (h) Leukemia

(i) Lung (j) Lymphoma (k) MLL (l) Prostate

(m) SRBCT

FIGURE 5. Analyzing the convergence of differential evolution-driven wrapper-based search across thirteen datasets.

fewer features across most datasets compared to state-of-the-art methods. When compared to rMRMR-MGWO, our approach consistently outperforms in all metrics except for the SRBCT dataset, where rMRMR-MGWO shows similar accuracy. Comparing our approach to QOMOJaya regarding feature subsets, our method outperforms in all cases as QOMOJaya does not significantly reduce the feature set. While QOMOJaya achieves similar ACC in four datasets, the number of features it selects is notably larger. MIM-mMFA outperforms our approach only in the *Braintumor* dataset in terms of ACC, while it shows similar ACC in the remaining four cases. However, it also does not significantly reduce the feature set. Similarly, BTLBOGSA performs better only for the Lung dataset but does not excel in reducing features or improving ACC in any aspects compared to our proposed approach. mRMR-MBFA shows similar ACC to our approach only for the *Leukemia* dataset, failing to produce better accuracy or feature reduction in the other three datasets. MBO-BLS does not outperform our approach in reducing the number of features or increasing ACC in any cases. BCOOTCSA and CFC-FBBA demonstrate similar ACC to our approach for two and four datasets, respectively. However, neither of them shows effectiveness in reducing the number of features. Finally, we compared our approach to MFI-RFPA. While they excel in reducing the number of selected features, they consistently achieve significantly lower ACC across all datasets, approximately 4%-5% less than our approach.

Based on the aforementioned analysis, it is evident that our approach not only enhances ACC but also excels in reducing features. When compared to state-of-the-art methods, it is notable that while some approaches achieve high classification performance for HDLSS datasets, they struggle with reducing the number of features. Conversely, methods effective at feature reduction often compromise on achieving higher ACC. The ideal approach should excel

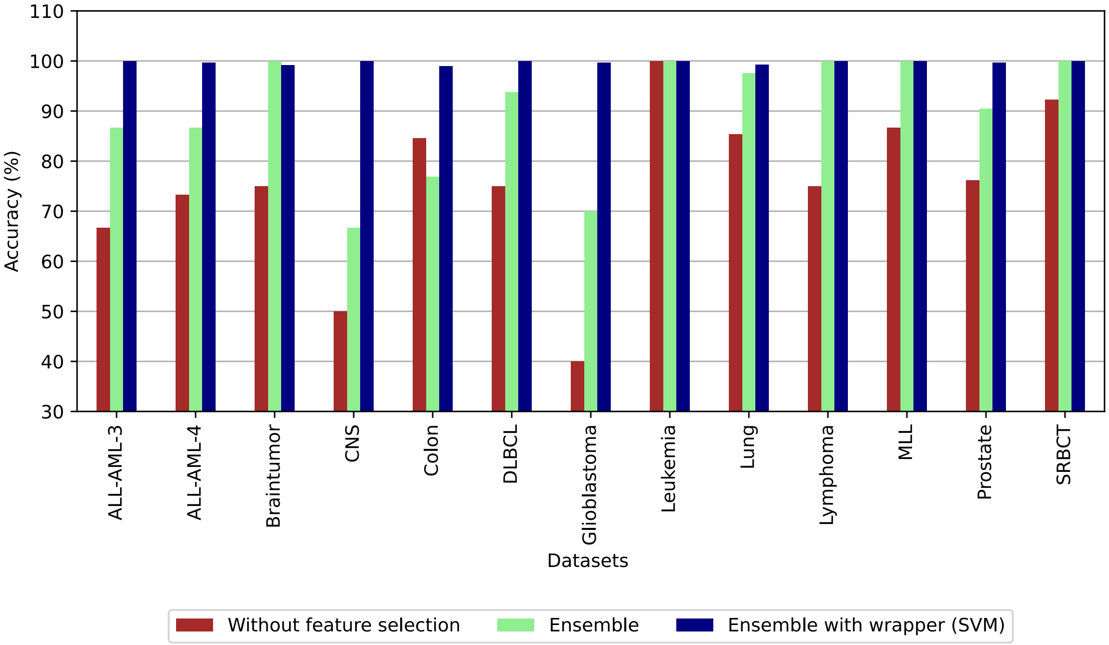


FIGURE 6. ACC with different approaches.

TABLE 8. Comparison in terms of ACC and number of selected features by the proposed method against state-of-the-art approaches.

Dataset	Metrics	rMRMR-MGWO	QOMO Jaya	MIM-mMFA	mRMR-MBFA	MBO-BLS	BCOOT-CSA	CFC-FBBA	BTL BOGSA	MFI-RFPA	Proposed (with SVM classif-caion)
ALL-AML-3	ACC	99	**100**	-	-	99.78	-	-	-	-	**100.00**
	# F	7	57.9	-	-	4	-	-	-	-	**3.90**
ALL-AML-4	ACC	98	**100**	-	-	-	-	-	-	-	99.72
	# F	11	59.6	-	-	-	-	-	-	-	**5.85**
Braintumor	ACC	-	-	**100**	-	-	95.42	93	96.9		99.17
	# F	-	-	11.93	-	-	12.66	9	15		**8.40**
CNS	ACC	97.39	99.99	-	-	-	93.22	-	-	95	**100.00**
	# F	17	60.4	-	-	-	7	-	-	**9**	10.50
Colon	ACC	94.14	97	-	90.48	98.2	94.75	98.83	98.87	95	**98.95**
	# F	10	13.6	-	9.4	5	8.75	1.8	16	**3**	5.70
DLBCL	ACC	-	-	-	98.75	-	-	**100**	99.62	91	**100.00**
	# F	-	-	-	5.98	-	-	5.7	17	**2**	4.00
Leukemia	ACC	-	-	**100**	**100**	99.45	-	**100**	94.15	96	**100.00**
	# F	-	-	18.7	4.3	3	-	3	16	3	**2.55**
Lung	ACC	97.52	99.93	-	-	-	-	-	**99.61**	99	99.33
	# F	17	117.9	-	-	-	-	-	13	**3**	8.55
Lymphoma	ACC	-	99.98	**100**	-	99.68	**100**	**100**	-	94	**100.00**
	# F	-	11.9	6.5	-	3	2	4.37	-	4	**1.95**
MLL	ACC	99.99	**100**	**100**	-	-	**100**	**100**	-	96	**100.00**
	# F	8	109.1	33	-	-	5	9.4	-	**3**	3.35
Prostate	ACC	-	-	-	94.18	-	-	99.42	98.42	95	**99.71**
	# F	-	-	-	6.7	-	-	5.25	7	**2**	4.45
SRBCT	ACC	**100**	**100**	**100**	-	99.87	100	-	99.17	95	**100.00**
	# F	12	15.2	27.3	-	6	6.34	-	11	7	**3.00**

in both aspects–improving ACC while reducing feature count. In this context, our approach emerges as a promising feature selection method for HDLSS data classification. The effectiveness of our proposed approach can be attributed to several factors. Firstly, the ensemble feature selection approach is able to eliminate irrelevant features by leveraging decision-making from five filtering approaches. This method tends to outperform individual filter-based feature selection approaches in minimizing irrelevant features. Secondly, in the subsequent phase, a metaheuristic based on DE is employed to further reduce the feature number. This DE approach can maintains a balance between exploration and exploitation, converging towards the global optimum. The associated objective function also guide the algorithm in pinpointing the optimal feature subset. Furthermore, the participation of SVM in both selecting the feature subset and determining final ACC also contributes to the efficacy of the HDLSS data classification task.

VI. CONCLUSION AND FUTURE DIRECTION

The selection of an appropriate subset of features from HDLSS data is a critical aspect of data classification. This is due to the presence of numerous redundant and irrelevant features that often impede the construction of an effective learning model. In this study, we propose a hybrid model that integrates filter and wrapper-based strategies to identify suitable subsets of features for the HDLSS data classification task. Our approach is two-fold. In the first stage, we employ an ensemble feature selection technique that combines five filter methods to identify a candidate feature subset. The second stage utilizes a wrapper-based approach with the DE metaheuristic as the search strategy. A linear combination of the classification error rate and the ratio of selected features serves as the evaluation metric for the feature subset. The datasets with the optimized feature sets are then used to train classification models. We employ SVM, KNN, and LR algorithms for classification. Our experimental results demonstrate that the proposed ensemble with wrapper-based feature selection method successfully reduces the number of features while maintaining superior ACC and AUC for almost all datasets. We are optimistic that researchers dealing with HDLSS and other high-dimensional data will find our method highly beneficial. Future investigations may explore the use of alternative wrapper and filter-based techniques to enhance HDLSS data classification tasks. It would be insightful to examine how various classification algorithms and fitness functions influence the performance of HDLSS data classification while concurrently reducing the cardinality of feature subsets. Furthermore, employing a broader variety of HDLSS datasets could aid in determining the generality of a feature subset selection algorithm.

INDIVIDUAL CONTRIBUTIONS

The research project involved a collaborative effort among multiple authors. Ashis Kumar Mandal and Md. Nadim led the conceptualization, while the methodology and software development were undertaken by Ashis Kumar Mandal. Data collection and curation were carried out by Ashis Kumar Mandal, Md. Nadim, Hasi Saha, Tangina Sultana, and Md. Delowar Hossain. Ashis Kumar Mandal was responsible for the initial draft preparation, with all authors actively participating in the subsequent review and editing process. Furthermore, all authors have read and agreed to the published version of the manuscript.

REFERENCES

[1] Q. Hang, J. Yang, and L. Xing, "Diagnosis of rolling bearing based on classification for high dimensional unbalanced data," *IEEE Access*, vol. 7, pp. 79159–79172, 2019.

[2] M. Leclercq, B. Vittrant, M. L. Martin-Magniette, M. P. S. Boyer, O. Perin, A. Bergeron, Y. Fradet, and A. Droit, "Large-scale automatic feature selection for biomarker discovery in high-dimensional OMICs data," *Frontiers Genet.*, vol. 10, May 2019.

[3] V. B. Kraus, "Biomarkers as drug development tools: Discovery, validation, qualification and use," *Nature Rev. Rheumatol.*, vol. 14, no. 6, pp. 354–362, Jun. 2018, doi: 10.1038/s41584-018-0005-9.

[4] M. Binkhonain and L. Zhao, "A machine learning approach for hierarchical classification of software requirements," *Mach. Learn. Appl.*, vol. 12, Jun. 2023, Art. no. 100457.

[5] M. H. Kabir, S. Mahmood, A. Al Shiam, A. S. Musa Miah, J. Shin, and M. K. I. Molla, "Investigating feature selection techniques to enhance the performance of EEG-based motor imagery tasks classification," *Mathematics*, vol. 11, no. 8, p. 1921, Apr. 2023.

[6] S. Gupta, D. Aga, A. Pruden, L. Zhang, and P. Vikesland, "Data analytics for environmental science and engineering research," *Environ. Sci. Technol.*, vol. 55, no. 16, pp. 10895–10907, Aug. 2021.

[7] M. N. Ashtiani and B. Raahemi, "Intelligent fraud detection in financial statements using machine learning and data mining: A systematic literature review," *IEEE Access*, vol. 10, pp. 72504–72525, 2022.

[8] R. Boutaba, M. A. Salahuddin, N. Limam, S. Ayoubi, N. Shahriar, F. Estrada-Solano, and O. M. Caicedo, "A comprehensive survey on machine learning for networking: Evolution, applications and research opportunities," *J. Internet Services Appl.*, vol. 9, no. 1, p. 16, Jun. 2018, doi: 10.1186/s13174-018-0087-2.

[9] V. Pappu, O. P. Panagopoulos, P. Xanthopoulos, and P. M. Pardalos, "Sparse proximal support vector machines for feature selection in high dimensional datasets," *Expert Syst. Appl.*, vol. 42, no. 23, pp. 9183–9191, Dec. 2015.

[10] Y. Chen, J. Bi, and J. Z. Wang, "MILES: Multiple-instance learning via embedded instance selection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 12, pp. 1931–1947, Dec. 2006.

[11] L. Wang, Y. Wang, and Q. Chang, "Feature selection methods for big data bioinformatics: A survey from the search perspective," *Methods*, vol. 111, pp. 21–31, Dec. 2016. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1046202316302742

[12] M. Montazeri, "HHFS: Hyper-heuristic feature selection," *Intell. Data Anal.*, vol. 20, no. 4, pp. 953–974, Jun. 2016.

[13] M. Charikar, V. Guruswami, R. Kumar, S. Rajagopalan, and A. Sahai, "Combinatorial feature selection problems," in *Proc. 41st Annu. Symp. Found. Comput. Sci.*, Nov. 2000, pp. 631–640.

[14] J. Li, K. Cheng, S. Wang, F. Morstatter, R. P. Trevino, J. Tang, and H. Liu, "Feature selection: A data perspective," *ACM Comput. Surv.*, vol. 50, no. 6, pp. 1–45, Dec. 2017, doi: 10.1145/3136625.

[15] K. Rajwar, K. Deep, and S. Das, "An exhaustive review of the metaheuristic algorithms for search and optimization: Taxonomy, applications, and open challenges," *Artif. Intell. Rev.*, vol. 56, no. 11, pp. 13187–13257, Nov. 2023.

[16] F.-T. Lin, C.-Y. Kao, and C.-C. Hsu, "Applying the genetic approach to simulated annealing in solving some NP-hard problems," *IEEE Trans. Syst., Man, Cybern.*, vol. 23, no. 6, pp. 1752–1767, Dec. 1993.

[17] M. Gendreau, *An Introduction to Tabu Search*. Boston, MA, USA: Springer, 2003, pp. 37–54.

[18] K. S. Tang, K. F. Man, S. Kwong, and Q. He, "Genetic algorithms and their applications," *IEEE Signal Process. Mag.*, vol. 13, no. 6, pp. 22–37, Dec. 1996.

[19] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc. Int. Conf. Neural Netw.*, vol. 4, 1995, pp. 1942–1948.

[20] P. Agrawal, H. F. Abutarboush, T. Ganesh, and A. W. Mohamed, "Metaheuristic algorithms on feature selection: A survey of one decade of research (2009–2019)," *IEEE Access*, vol. 9, pp. 26766–26791, 2021.

[21] D. Tao, X. Tang, X. Li, and X. Wu, "Asymmetric bagging and random subspace for support vector machines-based relevance feedback in image retrieval," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 7, pp. 1088–1099, Jul. 2006.

[22] N. Almugren and H. Alshamlan, "A survey on hybrid feature selection methods in microarray gene expression data for cancer classification," *IEEE Access*, vol. 7, pp. 78533–78548, 2019.

[23] D. S. Guru, M. Suhil, L. N. Raju, and N. V. Kumar, "An alternative framework for univariate filter based feature selection for text categorization," *Pattern Recognit. Lett.*, vol. 103, pp. 23–31, Feb. 2018.

[24] R. Meiri and J. Zahavi, "Using simulated annealing to optimize the feature selection problem in marketing applications," *Eur. J. Oper. Res.*, vol. 171, no. 3, pp. 842–858, Jun. 2006.

[25] Y. Zhang, D.-W. Gong, X.-Z. Gao, T. Tian, and X.-Y. Sun, "Binary differential evolution with self-learning for multi-objective feature selection," *Inf. Sci.*, vol. 507, pp. 67–85, Jan. 2020, doi: 10.1016/j.ins.2019.08.040.

[26] E. Hancer, B. Xue, and M. Zhang, "Differential evolution for filter feature selection based on information theory and feature ranking," *Knowl.-Based Syst.*, vol. 140, pp. 103–119, Jan. 2018.

[27] A. B. Brahim and M. Limam, "Ensemble feature selection for high dimensional data: A new method and a comparative study," *Adv. Data Anal. Classification*, vol. 12, no. 4, pp. 937–952, Dec. 2018.

[28] C.-F. Tsai and Y.-T. Sung, "Ensemble feature selection in high dimension, low sample size datasets: Parallel and serial combination approaches," *Knowl.-Based Syst.*, vol. 203, Sep. 2020, Art. no. 106097.

[29] L. Shen, M. J. Er, and Q. Yin, "Classification for high-dimension low-sample size data," *Pattern Recognit.*, vol. 130, Oct. 2022, Art. no. 108828.

[30] B. Liu, Y. Wei, Y. Zhang, and Q. Yang, "Deep neural networks for high dimension, low sample size data," in *Proc. 26th Int. Joint Conf. Artif. Intell.*, Aug. 2017, pp. 2287–2293.

[31] S. Chowdhury, X. Dong, and X. Li, "Recurrent neural network based feature selection for high dimensional and low sample size micro-array data," in *Proc. IEEE Int. Conf. Big Data*, Dec. 2019, pp. 4823–4828.

[32] A. Ç. Pehlivanlı, "A novel feature selection scheme for high-dimensional data sets: Four-staged feature selection," *J. Appl. Statist.*, vol. 43, no. 6, pp. 1140–1154, Apr. 2016.

[33] A. Bommert, X. Sun, B. Bischl, J. Rahnenführer, and M. Lang, "Benchmark for filter methods for feature selection in high-dimensional classification data," *Comput. Statist. Data Anal.*, vol. 143, Mar. 2020, Art. no. 106839.

[34] I. Jain, V. K. Jain, and R. Jain, "Correlation feature selection based improved-binary particle swarm optimization for gene selection and cancer classification," *Appl. Soft Comput.*, vol. 62, pp. 203–215, Jan. 2018.

[35] H. Rao, X. Shi, A. K. Rodrigue, J. Feng, Y. Xia, M. Elhoseny, X. Yuan, and L. Gu, "Feature selection based on artificial bee colony and gradient boosting decision tree," *Appl. Soft Comput.*, vol. 74, pp. 634–642, Jan. 2019.

[36] T. Thaher, A. A. Heidari, M. Mafarja, J. S. Dong, and S. Mirjalili, *Binary Harris Hawks Optimizer for High-Dimensional, Low Sample Size Feature Selection*. Singapore: Springer, 2020, pp. 251–272.

[37] S. S. Hameed, W. H. Hassan, L. A. Latiff, and F. F. Muhammadsharif, "A comparative study of nature-inspired metaheuristic algorithms using a three-phase hybrid approach for gene selection and classification in high-dimensional cancer datasets," *Soft Comput.*, vol. 25, no. 13, pp. 8683–8701, Jul. 2021.

[38] M. Ghosh, R. Guha, R. Sarkar, and A. Abraham, "A wrapper-filter feature selection technique based on ant colony optimization," *Neural Comput. Appl.*, vol. 32, no. 12, pp. 7839–7857, Jun. 2020.

[39] L. P. Cavalheiro, S. Bernard, J. P. Barddal, and L. Heutte, "Random forest kernel for high-dimension low sample size classification," *Statist. Comput.*, vol. 34, no. 1, p. 9, Oct. 2023, doi: 10.1007/s11222-023-10309-0.

[40] J. R. Choudhury, A. Saha, S. Roy, and S. Dutta, "Robust classification of high-dimensional data using data-adaptive energy distance," in *Proc. Joint Eur. Conf. Mach. Learn. Knowl. Discovery Databases*, Springer, 2023, pp. 86–101.

[41] G. Yang, W. Li, W. Xie, L. Wang, and K. Yu, "An improved binary particle swarm optimization algorithm for clinical cancer biomarker identification in microarray data," *Comput. Methods Programs Biomed.*, vol. 244, Feb. 2024, Art. no. 107987.

[42] A. Yaqoob, N. K. Verma, and R. M. Aziz, "Optimizing gene selection and cancer classification with hybrid sine cosine and cuckoo search algorithm," *J. Med. Syst.*, vol. 48, no. 1, p. 10, Jan. 2024.

[43] M. Vatankhah and M. Momenzadeh, "Self-regularized lasso for selection of most informative features in microarray cancer classification," *Multimedia Tools Appl.*, vol. 83, no. 2, pp. 5955–5970, Jan. 2024.

[44] P. E. Dhas, S. Lalitha, A. Govindaraj, and B. Jyoshna, "Spatial clustering based gene selection for gene expression analysis in microarray data classification," *Automatika*, vol. 65, no. 1, pp. 152–158, Jan. 2024.

[45] M. Rajab and D. Wang, "Practical challenges and recommendations of filter methods for feature selection," *J. Inf. Knowl. Manage.*, vol. 19, no. 1, Mar. 2020, Art. no. 2040019.

[46] H. O. Lancaster and E. Seneta, *Chi-Square Distribution*. Hoboken, NJ, USA: Wiley, 2005.

[47] A. S. Manek, P. D. Shenoy, M. C. Mohan, and K. R. Venugopal, "Aspect term extraction for sentiment analysis in large movie reviews using Gini index feature selection method and SVM classifier," *World Wide Web*, vol. 20, no. 2, pp. 135–154, Mar. 2017.

[48] Y.-W. Chen and C.-J. Lin, *Combining SVMs With Various Feature Selection Strategies*. Berlin, Germany: Springer, 2006, pp. 315–324.

[49] N. Veyrat-Charvillon and F.-X. Standaert, "Mutual information analysis: How, when and why?" in *Cryptographic Hardware and Embedded Systems—CHES 2009*, C. Clavier and K. Gaj, Eds. Berlin, Germany: Springer, 2009, pp. 429–443.

[50] Y. He, H. Yu, R. Yu, J. Song, H. Lian, J. He, and J. Yuan, "A correlation-based feature selection algorithm for operating data of nuclear power plants," *Sci. Technol. Nucl. Installations*, vol. 2021, Aug. 2021, Art. no. 9994340.

[51] T. O. Kvalseth, "Entropy and correlation: Some comments," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-17, no. 3, pp. 517–519, May 1987.

[52] K. V. Price, *Differential Evolution*. Berlin, Germany: Springer, 2013, pp. 187–214.

[53] Y. He, F. Zhang, S. Mirjalili, and T. Zhang, "Novel binary differential evolution algorithm based on taper-shaped transfer functions for binary optimization problems," *Swarm Evol. Comput.*, vol. 69, Mar. 2022, Art. no. 101022.

[54] R. Storn and K. Price, "Differential evolution—A simple and efficient heuristic for global optimization over continuous spaces," *J. Global Optim.*, vol. 11, no. 4, pp. 341–359, Dec. 1997.

[55] O. A. Alomari, S. N. Makhadmeh, M. A. Al-Betar, Z. A. A. Alyasseri, I. A. Doush, A. K. Abasi, M. A. Awadallah, and R. A. Zitar, "Gene selection for microarray data classification based on gray wolf optimizer enhanced with TRIZ-inspired operators," *Knowl.-Based Syst.*, vol. 223, Jul. 2021, Art. no. 107034.

[56] A. Chaudhuri and T. P. Sahu, "Multi-objective feature selection based on quasi-oppositional based Jaya algorithm for microarray data," *Knowl.-Based Syst.*, vol. 236, Jan. 2022, Art. no. 107804.

[57] A. Dabba, A. Tari, S. Meftali, and R. Mokhtari, "Gene selection and classification of microarray data method based on mutual information and moth flame algorithm," *Expert Syst. Appl.*, vol. 166, Mar. 2021, Art. no. 114012.

[58] W. Xie, L. Wang, K. Yu, T. Shi, and W. Li, "Improved multi-layer binary firefly algorithm for optimizing feature selection and classification of microarray data," *Biomed. Signal Process. Control*, vol. 79, Jan. 2023, Art. no. 104080.

[59] P. Parhi, R. Bisoi, and P. K. Dash, "Influential gene selection from high-dimensional genomic data using a bio-inspired algorithm wrapped broad learning system," *IEEE Access*, vol. 10, pp. 49219–49232, 2022.

[60] E. Pashaei and E. Pashaei, "Hybrid binary COOT algorithm with simulated annealing for feature selection in high-dimensional microarray data," *Neural Comput. Appl.*, vol. 35, no. 1, pp. 353–374, Jan. 2023.

[61] A. Esfandiari, F. Farivar, and H. Khaloozadeh, "Fractional-order binary bat algorithm for feature selection on high-dimensional microarray data," *J. Ambient Intell. Humanized Comput.*, vol. 14, no. 6, pp. 7453–7467, Jun. 2023.

[62] A. K. Shukla, P. Singh, and M. Vardhan, "Gene selection for cancer types classification using novel hybrid metaheuristics approach," *Swarm Evol. Comput.*, vol. 54, May 2020, Art. no. 100661.

[63] M. Li, L. Ke, L. Wang, S. Deng, and X. Yu, "A novel hybrid gene selection for tumor identification by combining multifilter integration and a recursive flower pollination search algorithm," *Knowl.-Based Syst.*, vol. 262, Feb. 2023, Art. no. 110250.

ASHIS KUMAR MANDAL received the B.Sc.Eng. degree in computer science and engineering from the Shahjalal University of Science and Technology, Bangladesh, in 2007, the Master of Computer Science degree from University Malaysia Pahang (UMP), Malaysia, in 2017, and the Ph.D. degree in software and information science from Iwate Prefectural University, Iwate, Japan, in 2022. He is currently a Postdoctoral Research Fellow with the Computer Science Department, University of Saskatchewan, Canada. He is also a Professor with the Computer Science and Engineering Department, Hajee Mohammad Danesh Science and Technology University, Bangladesh. He has more than 14 years of research and teaching experience in the domain of computer and information science. His research interests include machine learning, quantum computing, and software engineering. He was awarded the Japanese Government (Monbukagakusho: MEXT) Scholarship for the Ph.D. studies. He received the ICT Fellowship from the Government of the People's Republic of Bangladesh for the Master of Computer Science degree. He was also awarded the Silver Medal at the Citrex Research Exposition, UMP, Malaysia, in 2015, and the President's Award from Iwate Prefectural University, in 2022, for his research achievements.

MD. NADIM received the B.Sc. degree in computer science and engineering from Hajee Mohammad Danesh Science and Technology University, Dinajpur, Bangladesh, in 2010, and the M.Sc. degree in computer science from the University of Saskatchewan, Canada, in September 2018, where he is currently pursuing the Ph.D. degree. In 2012, he became a Lecturer with Hajee Mohammad Danesh Science and Technology University, kicking off his academic career, and he upgraded to an Assistant Professor, in 2015. His Ph.D. research explores bug-inducing source code patterns, software commits activities, developers' coding behaviors, and the use of advanced automatic machine learning/deep learning algorithms to classify these characteristics effectively. He is also highly fascinated by contributing to the use and improvement of quantum computing techniques, specifically quantum machine learning (QML) algorithms in the domain of software engineering research. He aims to bridge the gap between traditional software development methods and innovative technologies, making significant contributions to the rapidly evolving field of computer science and software engineering. His academic journey highlights an expedition for knowledge and dedication to pushing the boundaries of research and technology, leaving a lasting impact on both academic and industrial fronts.

HASI SAHA received the B.Sc. degree in CSE from Hajee Mohammad Danesh Science and Technology University, Dinajpur, and the master's degree in IT from the University of Dhaka. She is currently an Associate Professor with the Department of Computer Science and Engineering, Hajee Mohammad Danesh Science and Technology University. Her research interests include machine learning, deep learning, natural language processing, artificial intelligence, and the Internet of Things.

TANGINA SULTANA received the B.Sc. degree from the Department of Telecommunication and Electronic Engineering, Hajee Mohammad Danesh Science and Technology University, Bangladesh, in 2010, the M.Sc. degree from the Institute of Information and Communication Technology, Bangladesh University of Engineering and Technology (BUET), Bangladesh, in 2015, and the Ph.D. degree in computer science engineering from Kyung Hee University, Republic of Korea, in 2022. She was a Postdoctoral Researcher with the Department of Computer Science and Engineering, Kyung Hee University. She is currently an Associate Professor with the Department of Electronics and Communication Engineering, Hajee Mohammed Danesh Science and Technology University. Her current research interests include graph mining, graph compression, wireless communications, edge computing, and machine learning. She was a recipient of the Gold Prize in KDBC 2020 and KDBC 2021, as well as KDBC 2023, and the Best Paper Award from IEEE BigComp 2021.

MD. DELOWAR HOSSAIN received the B.Sc. and M.Sc. degrees from the Department of Information and Communication Engineering (ICE), Islamic University, Kushtia, Bangladesh, in 2004 and 2005, respectively, and the Ph.D. degree from the Department of Computer Science and Engineering, Kyung Hee University, South Korea, in 2022. He was a Visiting Scholar with Infosys, Bengaluru, India. He was a Postdoctoral Research Fellow with the Intelligent Computing and Security Laboratory, Kyung Hee University. He is currently a Professor with the Computer Science and Engineering Department, Hajee Mohammad Danesh Science and Technology University, Bangladesh. He specializes in the analysis and optimization of edge computing networks and the IoT systems using game theory and machine learning approaches. His current research interests include cloud/edge/fog computing, vehicular edge computing, big data, machine learning, and the Internet of Things. He received the Best Paper Award from KSC 2018, KSC 2019, and KCC 2021, South Korea.

EUI-NAM HUH (Member, IEEE) received the master's degree in computer science and engineering from The University of Texas at Arlington, USA, in 1995, and the Ph.D. degree from Ohio University, USA, in 2002.

Since 2023, he has been the Director of the ICT Research Center (ITRC) funded from national project subjected to cloud continuum for eight years. He is currently a Professor with the Department of Computer Science and Engineering, Kyung Hee University, South Korea. He was the Chair of the Department and the Dean of Information Administration with Kyung Hee University for 7.5 years. His research interests include cloud computing, the Internet of Things, future internet, distributed real-time systems, mobile computing, big data, and security. He serves on the review board for the National Research Foundation of Korea. He has actively participated in community services for several organizations, including ITU-T SG13, *Applied Sciences*, ICCSA, WPDRTS/IPDPS, APAN Sensor Network Group, ICUIMC, IMCOM, ICONI, APIC-IST, ICUFN, and SoICT, as different types of editors, and various chair positions.