

Heterogeneous Ensemble Feature Selection Model (HEFSM) for Big Data Analytics

M. Priyadharsini^{1,*} and K. Karuppasamy²

¹Department of Information Technology, Sri Ramakrishna Engineering College, Coimbatore, Tamilnadu, India

²Department of Computer Science & Engineering, RVS College of Engineering and Technology, Coimbatore, Tamilnadu, India

*Corresponding Author: M. Priyadharsini. Email: priya.murugesan@srec.ac.in

Received: 11 April 2022; Accepted: 13 June 2022

Abstract: Big Data applications face different types of complexities in classifications. Cleaning and purifying data by eliminating irrelevant or redundant data for big data applications becomes a complex operation while attempting to maintain discriminative features in processed data. The existing scheme has many disadvantages including continuity in training, more samples and training time in feature selections and increased classification execution times. Recently ensemble methods have made a mark in classification tasks as combine multiple results into a single representation. When comparing to a single model, this technique offers for improved prediction. Ensemble based feature selections parallel multiple expert's judgments on a single topic. The major goal of this research is to suggest HEFSM (Heterogeneous Ensemble Feature Selection Model), a hybrid approach that combines multiple algorithms. The major goal of this research is to suggest HEFSM (Heterogeneous Ensemble Feature Selection Model), a hybrid approach that combines multiple algorithms. Further, individual outputs produced by methods producing subsets of features or rankings or voting are also combined in this work. KNN (K-Nearest Neighbor) classifier is used to classify the big dataset obtained from the ensemble learning approach. The results found of the study have been good, proving the proposed model's efficiency in classifications in terms of the performance metrics like precision, recall, F-measure and accuracy used.

Keywords: PSO (Particle Swarm Optimization); GWO (Grey Wolf Optimization); EHO (Elephant Herding Optimization); data mining; big data analytics; feature selection; HEFSM classifier

1 Introduction

IS (Intelligent Systems) have been growing rapidly in recent years encompassing the areas of commerce, science and medicine [1]. DM (Data Mining) defined in short as knowledge explorations identify trends, patterns and relationships in data and convert raw data into meaningful information [1]. AI (Artificial Intelligence) based techniques have been useful in normalizing incomplete DM datasets. DMTs (DM techniques) have been exploited in business decisions and organizational priorities. DM tasks include pre-



This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

processing, pattern assessments and presentations [2] using Cluster analysis, Time-series mining, ARM (Association Rules Mining) and classifications [3]. Thus, classification techniques play the underlying role in all DM tasks. Classifications grow in complexity when it involves high dimensional data like Big data where dimension is referred to the number of parameter values taken into consideration. As this count increases it increases the computational time required for processing where dimensionality reductions of the data is executed to increase the speed of the processing techniques.

Feature Selection is an efficient method of dimensionality reductions [4]. These selections select a subset with required features from the base set based on an evaluation criteria. The results thus better learning performances in terms of accuracy or classifications. These selections search for minimal representation of features and evaluate thus selected subsets in classification performances [5]. Feature selection methods also contribute towards generalizations of classifiers [6]. These approaches are oriented towards redundancy minimizations and relevance maximizations of labelling classes for classifications. The selected feature subsets retain their original forms and offer better readability and interpretability. When information is labelled, it helps in distinguishing relevant features. Thus these methods can be categorized into supervised, unsupervised and semi-supervised methods for feature selections [7].

Supervised Feature Selections can be viewed as filters, wrappers and embeddings in modelling systems [8,9]. Filters separate unwanted features for classifiers to learn while wrappers use predictive accuracies of previous learning for selecting quality features in data sets. Wrappers are expensive in their executions where metaheuristic algorithms reduce these computational times reasonably. Most metaheuristic algorithms have a problem of getting caught in a local optimum. Evolutionary algorithms have helped overcome this problem in their operations where PSO, GA (Genetic Algorithm), DD (Differential Development), ACO (Ant Colony Optimization), GWO and EHO algorithms are examples in this category. Hybrid techniques aim to use individual algorithmic powers for increased explorations [10].

This paper proposes a new hybrid wrapper for feature selection using FWGWO (Fuzzy Weight Grey Wolf Optimization) and EHO algorithms. Features are selected based on a fuzzy weight that minimizes the subset's length and increases classification accuracy in parallel. The proposed HEFSM uses multiple algorithms. TCM (Tent Chaotic Map) is the wrapper used to jump local optima while optimizing search capabilities. FWGWO is an easy to implement algorithm which operates with fewer parameters. The algorithm considers location of individual wolves for its solutions. Thus, this work uses EHO in updating positional locations. A binary transformation normalizes continuous features which are then classified by the KNN classifier.

2 Literature Review

Consider Sharawi et al. [11] used WOA (Whale Optimization Algorithm) in their proposed feature selection system. Their meta-heuristic optimization algorithm mimicked the behavior of humpback whales. The proposed model applies the wrapper-based method to reach the optimal subset of features. They compared their technique with the PSO and GA algorithms on sixteen different datasets with multiple parameters. Experimental results demonstrate the proposed algorithm's ability to optimize feature selections.

Sayed et al. [12] embedded a chaotic search in WOA iterations. Their scheme called CWOA (Chaotic Whale Optimization Algorithm) used TCM's for improving WOA outputs. Their chaotic exploration operators outperformed other similar techniques. PSO was proposed by Chen et al. [13] in their HPSO-SSM (Hybrid Particle Swarm Optimization with a Spiral-Shaped Mechanism) selected optimal feature subsets for classification using wrappers. Their experimental results showed good performance in searching feature feasibility by selecting informative attributes for classifications.

PSO variant was used by Gu et al. [14] in their study which proposed CSO (Competitive Swarm Optimize) for large-scale optimizations of features in high-dimensional data. Their experimental results of six benchmark datasets showed a better performance than PSO based algorithms. Their CSO selected lesser number of features which could be classified better. Tu et al. [15] proposed HSGWO (Hierarchy Strengthened GWO) as their feature selection technique where wolves were classified as omega and dominant. The technique's learning used dominant wolves to prevent low ranked wolves and improve cumulative efficiency. This is followed by a hybrid GWO and DE (Differential Evolution) strategy for omega wolves to evade local optimums. Their use of a perturbed operator improved exploration of the diverse population. Their results proved the algorithm's superiority in convergences and solution quality.

Dimensions were learnt by Haiz et al. [16] with a generalized learning on PSO selected features. They included cardinality of subsets by extending dimension velocity. Their 2D-learning framework retained key features of PSO selections while learning dimensions. When evaluated with NB (Naive-Bayes) and KNN classifications their feature subsets ran faster evaluating its usefulness.

GWO was combined with Antlion Optimization in the study of Zawbaa et al. [17]. The scheme called ALO-GWO (Antlion Optimization -Grey Wolf Optimization) algorithm learnt from fewer examples and used it to select important features from large sets that could improve classification accuracy. The proposal showed promising results in comparison with PSO and GA. APSO (Accelerated Particle Swarm Optimization) proposed by Fong et al. [18] was a lightweight feature selection technique. It was designed specifically for mining streamed data for achieving enhanced analytical accuracy with lesser processing time. The scheme when tested on high dimensionality sets showed its improved performance.

Two-phase Mutation figured in the study of Abdel-Basset et al. [19] based on GWO algorithm. They selected features with wrappers. Their sigmoid function transformed search spaces into a binary form for feature selections. Statistical analysis proved the technique's effectiveness. Li et al. [20] integrated GWO with a Kernel Extreme Learning Machine in their study. Their framework called (IGWO-KELM) was applied in medical diagnostics. GWO and GA were evaluated in its comparisons for predicting common diseases. Their performance metrics showed improvements in classification accuracy, sensitivity, specificity, precision, G-mean, F-measure and selected feature sizes. GWSO's binary variant was exploited by ElHasnony et al. [21] in their study. Their technique called GWO-PSO found optimal solutions and used KNN for classifications on wrapper selected features where Euclidean separation matrices found optimal solutions.

Wang et al. [22] proposed a new technique SA-EFS (Sort aggregation-Ensemble Feature Selection) for high-dimensional datasets. The technique used Chi-Square, maximum information coefficient and XGBoost for aggregations. They integrated arithmetic and geometric mean aggregations for analyzing classification and predictive performances of the technique's selected feature subsets. Their experimental results showed arithmetic mean aggregation ensemble feature selections effectively improved classification accuracy and that threshold intervals of 0.1 performed better. An empirical study in Pes [23] included many types of selection algorithms in different application domains. Eighteen heterogeneous classification tasks were evaluated with different cardinality feature subsets where ensemble approaches performed better even on single selectors.

Despite numerous attempts to develop an efficient model for characteristic choice in Big Data applications, the complexity of handling such data remains a significant barrier. As a result of the large volume and intricacy of large data sets, the data mining operation may be hampered. The characteristic choice method is a required pre-processing phase to reduce dataset dimensionality for the most useful characteristics and categorization performance improvement. It takes time to conduct a comprehensive analysis for important traits. In this paper, a new Heterogeneous Ensemble Feature Selection Model)

which includes several algorithms in a hybrid combination is proposed. To determine the best answers, a K-nearest neighbour classifier with Euclidean separation matrices is utilised.

3 Proposed HEFSM Methodology

This work proposes feature selections based on sorted aggregations using ensembles in its approach. Three feature selection methods are executed in HEFSM to obtain multiple optimal feature subsets. Filtering is done by ReliefF, InfoG (Information Gain), and GR (Gain Ratio) methods. The proposed work also embeds SVM-t (Support Vector Machine- t-statistics). A new binary variant of wrappers based on the FWGWO and EHO is used in feature selections. Individual outputs of the used methods are learnt based on rules for getting multiple optimal feature subset candidate sets which are then aggregated to produce optimized feature subsets. The proposed algorithm of this work is verified using KNN classification. Fig. 1 depicts the flow of this research work.



Figure 1: Flow of the proposed system

3.1 HEFSM

Ensembles learn better in ML by combining different ML learning models [24,25]. They are better than singular ML models as they effectively output multiple optimal features. Hence, this work uses ensembles for better feature selections by using combinations of filters, wrappers and an embedded algorithm.

3.1.1 Filters

Data Filtering in this work are based on three functions namely ReliefF, InfoG, and GR [25].

ReliefF

ReliefF differentiates data points that are near to each other in a feature space. A sample drawn instance's value is compared with instances near to it. A relevance score is then assigned to each feature as good features have a constant value within their class. This process is recursive and features' scores in iterations are updated. This helps weighing nearest neighbours based on their distance. ReliefF which can handle noisy and incomplete data values is limited to two classes in this work. It randomly selects an instance R_i and searches for k nearest neighbors in the same class (nearest hits H_j) and different classes (misses $M_j(C)$). The quality estimation of features F is updated based on R_i , H_j and $M_j(C)$. Each class's misses are weighted with the prior probability of that class $P(C)$. Hits and misses are computed in the interval $[0,1]$ where the probability weights total to 1. Misses are divided by probability weight factor $1 - P(\text{class}(R_i))$ or sum of probabilities of misses. This process is repeated m times. The selection of user defined k for hits and misses ensures robustness with respect to noises in the data where k controls estimates and is generally set to 10 for safety.

Algorithm 1: ReliefF Algorithm

Input: For each training instance a vector of feature values and the class value

Output: The vector W of estimations of the qualities of features

1. Set all weights $W[F] := 0$;
 2. For $i := 1$ to m do begin
 3. Randomly select an instance R_i
 4. Find k nearest Hits H_j
 5. For each Class $C \neq \text{class}(R_i)$ do
 6. From class C find k nearest misses $M_j(C)$;
 7. For $F := 1$ to a do
 8. $W[F] := W[F] - \sum_{j=1}^k \frac{\text{diff}(F, R_i, H_j)}{(m.k)} + \sum_{C \neq \text{class}(R_i)} \left[\frac{P(C)}{1 - P(\text{class}(R_i))} \sum_{j=1}^k \frac{\text{diff}(F, R_i, M_j(C))}{(m.k)} \right]$
 9. End
-

InfoG

InfoG is based on entropy where the weight of each feature is derived by evaluating the extent to which it decreases the class entropy or in reducing class predictions. Entropy can be calculated [25] using Eq. (1):

$$IG(C, F) = H(C) - H(C|F) \quad (1)$$

where $IG(C, F)$ -information of feature F in class C , $H(C)$ -class entropy and $H(C|F)$ -conditional class entropy. InfoG is the number of bits saved when the class is transformed. Conditional entropy, depicted in Eqs. (2) and (3) can be computed by splitting the dataset into groups for each observed value of F and the sum of the ratio of examples in the class multiplied by the group's entropy [25].

$$H(C|F) = - \sum_{c \in C} p(C|F) \log p(C|F) \quad (2)$$

$$H(C) = - \sum_{c \in C} p(C) \log p(C) \quad (3)$$

$H(C|F)$ -conditional class entropy using Eq. (2), $H(C)$ -class entropy using Eq. (3).

GR

GR is also based on entropy, but differs from InfoG in normalizing InfoG's bias towards greater values. Normalizations are done by dividing InfoG with attribute entropy in a class; as a result it reduces the bias of InfoG algorithm. GR can be computed using Eq. (4) [26].

$$\text{Gain Ratio (C, F)} = (H(C) - H(C|F))/H(F) \quad (4)$$

when features count in a subset increases, difference are more clear and pronounced.

Wrappers

Wrappers use ML for their search of possible feature subsets and evaluate each subset based on quality. This work uses FWGWO (Algorithm 1) with EHO (Algorithm 2) as a wrapper in its operations.

FWGWO

Techniques based on Grey wolves are an advanced implementation of optimizations as they easily close on optimal selection of features (prey) from a dataset. This work uses metaheuristic algorithms which mimic attacking grey wolves for closing in on a prey (optimal selection of features) [27]. Wolves exist in packs of 5–12 and have four levels of leadership hierarchy namely alpha (α), beta (β), delta (δ), and omega (ω) as shown in Fig. 2. Alpha represents pack leaders in decision making.

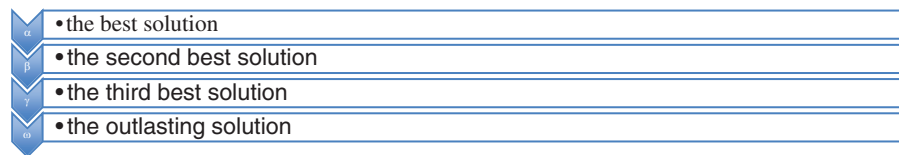


Figure 2: The leadership hierarchy of wolfs

The α , β , and δ are the first, second, third best-recommended optimal features from the dataset in FWGWO algorithm. The best estimated solution in feature elections is ω . In a hunting process, wolves surround their prey in an arrangement that can be depicted mathematically using Eqs. (5) and (6) (5–6) [28]:

$$\vec{D} = |\vec{C} \cdot \vec{X}_P - \vec{X}_{(t)}| \quad (5)$$

$$\vec{X}_{(t+1)} = |\vec{X}_{P(t)} - \vec{A} \cdot \vec{D}| \quad (6)$$

where $\vec{X}_{(t)}$ and $\vec{X}_{(p)}$ -locations of grey wolf and prey in iteration (t) for selecting features. The coefficient vectors \vec{A} and \vec{C} in the above equations can be modelled using Eqs. (7) and (8),

$$\vec{A} = |2 \vec{a} \cdot \overrightarrow{rand1} - \vec{a}| \quad (7)$$

$$\vec{C} = 2 \cdot \overrightarrow{rand2} \quad (8)$$

where $\overrightarrow{rand1}$ and $\overrightarrow{rand2}$ - random vectors in the interval [0,1] in feature selections. \vec{a} is linearly decreased from 2 to 0 by iterations using Eq. (9),

$$a(t) = a_s - (a_s - a_f) * \left(\frac{t}{tmax}\right)^2 \quad (9)$$

where a_s and a_f -initial and end value of a control parameter, t-current iteration and Tmax-overall iterations. The iterations parameter controls iterations during feature selection. A grey wolf's location (X, Y) changes

based on its prey position (X^*, Y^*) in selecting features. α , β , and δ mimic hunting behavior for feature selections. Since, three solutions result, remaining wolves change their positions based on three best solutions $(\vec{X}_1, \vec{X}_2$ & $\vec{X}_3)$ and shown as Eqs. (10)–(13).

$$\vec{X}_{(t+1)} = \frac{(\vec{X}_1 \cdot we_1 + \vec{X}_2 \cdot we_2 + \vec{X}_3 \cdot we_3)}{3} \tag{10}$$

$$\vec{X}_1 = \vec{X}_\alpha - \vec{A}_1 \cdot (\vec{D}_\alpha) \tag{11}$$

$$\vec{X}_2 = \vec{X}_\beta - \vec{A}_2 \cdot (\vec{D}_\beta) \tag{12}$$

$$\vec{X}_3 = \vec{X}_\delta - \vec{A}_3 \cdot (\vec{D}_\delta) \tag{13}$$

The weight vector $WE = we_1(\alpha) + we_2(\beta) + we_3(\delta)$ is obtained and its membership function is estimated using interval weights of interval/crisp comparison matrix. If F_w is a fuzzy membership function for a fuzzy set A on the universe of discourse w it is defined as $\mu_A: F_w \rightarrow [0,1]$, where each element of w is in they interval $[0,1]$, $w = \{we_1, we_2, we_3\}$ and $we_1 \geq we_2 \geq we_3$. Mathematically, in a search alpha’s weights changes to $we_1 = \{1.0 \text{ to } \frac{1}{3}\}$ while the weights of the beta and delta change to $\{0.0 \text{ to } \frac{1}{3}\}$. $y = \text{pimf}(w, \text{params})$ returns fuzzy membership values computed using a spline-based pi-shaped membership function as depicted in Eq. (14),

$$F_w = \mu_A(w, a, b, c, d) = \begin{cases} 0, & w \leq a \\ 2\left(\frac{w-a}{b-a}\right)^2, & a \leq w \leq \frac{a+b}{2} \\ 1 - 2\left(\frac{w-b}{b-a}\right)^2, & \frac{a+b}{2} \leq w \leq c \\ 1, & b \leq w \leq c \\ 1 - 2\left(\frac{w-c}{d-c}\right)^2, & c \leq w \leq \frac{c+d}{2} \\ 2\left(\frac{w-d}{d-c}\right)^2, & \frac{c+d}{2} \leq w \leq d \\ 0, & w \geq d \end{cases} \tag{14}$$

Membership values for each weight in $w = \{we_1, we_2, we_3\}$ is computed using Eq. (15).

$$\vec{X}_{(t+1)} = \frac{(\vec{X}_1 \cdot F_{we_1} + \vec{X}_2 \cdot F_{we_2} + \vec{X}_3 \cdot F_{we_3})}{3} \tag{15}$$

where, $F_{we_1}, F_{we_2}, F_{we_3}$ -weights of α, β , and δ .

$$\vec{D}_\alpha = |\vec{C}_1 \cdot \vec{X}_\alpha - \vec{X}| \tag{16}$$

$$\vec{D}_\beta = |\vec{C}_2 \cdot \vec{X}_\beta - \vec{X}| \tag{17}$$

$$\vec{D}_\delta = |\vec{C}_3 \cdot \vec{X}_\delta - \vec{X}| \tag{18}$$

Wolves change their positions for catching a prey where $|A| < 1$. α, β , and δ wolves pursue the prey and unite on closing. \vec{A}_1 has takes values > 1 or < -1 randomly.

Algorithm 2: FWGWO algorithm

Input ← grey wolf population X, number of iteration Tmax, size N, swing factor C, control parameter a, and learning factors c1, c2

Output ← Optimal solution (α)

1. Begin
2. **For** $t = 1: Tmax$
3. **For** $i = 1: N$
4. **For** $j = 1: d$
5. Calculate $\vec{D}_\alpha, \vec{D}_\beta, \vec{D}_\gamma$ according to Eqs. (16)–(18)
6. Calculate \vec{X}_1, \vec{X}_2 & \vec{X}_3 according to Eqs. (11)–(13)
7. Update position using Eq. (10)
8. End for
9. End for
10. Calculate fitness value and then update α, β , and δ
11. Update a according to Eq. (9)
12. Update A and C according to Eqs. (7),(8)
13. Update $\vec{X}_\alpha, \vec{X}_\beta$ & \vec{X}_γ
14. End for
15. End

Proposed Elephant Herding Optimization (EHO)-FWGWO Model

The proposed algorithm for addressing feature selection problems is examined in this section (see Fig. 3). The model is mainly initialized, evaluated, transformed, and iterated.

Initialization

In this stage randomly generated n wolves are initialized. These wolves or search agents represent a solution whose length d equates to the features count in the original dataset. Fig. 4 displays a probable dataset solution with 11 features. Relevant features have a value of 1 while ignored features take a value of 0.

A logistic map generated the initial PSO population where feature convergences in the proposed algorithm rely on random sequences that execute the algorithm using different parameters. The proposed work's TCM is depicted as Eq. (19).

$$x_{k+1} = \begin{cases} \mu x_k & \text{for } 0 \leq x_k \leq \frac{1}{2} \\ \mu(1 - x_k) & \text{for } \frac{1}{2} \leq x_k \leq 1 \end{cases} \quad (19)$$

where, μ -set to $1+R$, R-random value in the range $[0,1]$. A point x_0 in the interval is changed to x_k in the range $[0,1]$.

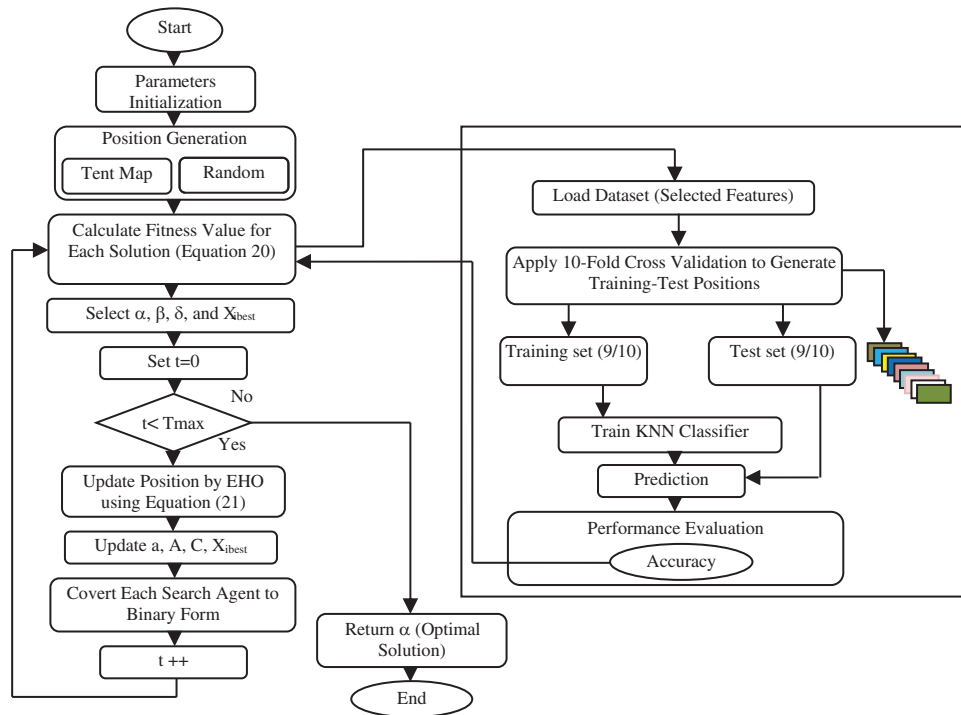


Figure 3: Block diagram for the proposed model

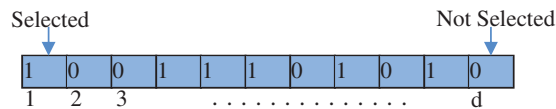


Figure 4: Solution representation

Evaluation

The fitness function used in this work evaluates goals using Eq. (20),

$$fitness = \alpha \gamma_R(D) + \beta \frac{|S|}{|D|} \tag{20}$$

where, $\gamma_R(D)$ -present the error rate of KNN, $|S|$ -selected subset feature cardinality (length), $|D|$ -cardinality of all dataset features. $\alpha \in [0, 1]$ and $\beta = 1 - \alpha$ - weights that reflect accuracy value of classifications and the selected sub-set length.

Position Updating

FWGWO considers the second and third-best locations of wolves for learning location information on specific wolves and in doing so a wolf’s information on own experience is ignored. Hence, the proposed work uses EHO to update positions. FWGWO update are performed based on a clan operator which uses grey wolf’s position in a clan for updates based on the relationship to the operator. EHO methods for

updates described by the authors of [29] are used in this study for updating grey wolf positions. Assuming c_i denotes a grey wolf clan its subsequent position in the clan is updated using Eq. (21):

$$X_{new,ci,j}(t+1) = c_1 r_2 (F_{we1} \vec{X}_1(t) + F_{we2} \vec{X}_2(t) + F_{we3} \vec{X}_3(t)) * X_{ci,j}(t) + a_1 * c_2 r_2 (X_{best,ci} - X_{ci,j}(t)) * r \quad (21)$$

where, $X_{new,ci,j}(t)$ -updated position at iteration t, $X_{ci,j}(t)$ -previous position of grey wolf j in the ci at iteration t. $X_{best,ci}$ -fittest wolf in clan ci; the fittest individual grey wolf. The scaling factor $a_1 \in [0, 1]$ influences matriarch of ci in $X_{ci,j}(t)$. $r \in [0, 1]$. This is a stochastic distribution which can assess diversity in the population during a search. This work does not use uniform distribution or $X_{ci,j} = X_{best,ci}$ implying the fittest grey wolf information cannot be updated using Eq. (21). This is overcome in this work by using EHO for updating individual fittest wolf position given by Eq. (22):

$$X_{new,ci,j}(t+1) = X_{center,ci}(t) \times a_2 \quad (22)$$

The issue in the above equation is the absence of a reference source. Hence, information from all individuals in ci is used to create the new best $X_{new,ci,j}$ where its centre $X_{center,ci}$, is calculated for the d^{th} dimension through D (Total Dimensions) calculations using Eq. (23):

$$X_{center,ci,d}(t+1) = \frac{1}{n_{ci}} \times \sum_{j=1}^{n_{ci}} X_{ci,j,d}(t) \quad (23)$$

where, $1 \leq d \leq D$ - d^{th} dimension, n_{ci} -count of individuals in ci, and $X_{ci,j,d}(t)$ - d^{th} dimension of the individual $X_{ci,j}$ at iteration t. Based on the feature selection values (0 or 1), most suitable features are chosen for enhancing accuracy of classifiers. The resultant search space is transformed by a sigmoidal function into a continuous binary form [19] using Eqs. (24) to (26),

$$X_i = \frac{x_i - Min}{Max - Min} \quad (24)$$

$$x_{s_i} = \frac{1}{1 + e^{-X_i}} \quad (25)$$

$$x_{binary} = \begin{cases} 0 & \text{if } R < x_{s_i} \\ 1 & \text{if } R \geq x_{s_i} \end{cases} \quad (26)$$

where, x_{s_i} -S-shaped search agent's continuous value (feature), $i = 1, \dots, d$, and x_{binary} - values to 0 or 1 using random number $R \in [0, 1]$. Max/Min are maximum/minimum values of the continuous feature vectors. Algorithm 3 lists the EHO-FWGWO algorithms pseudo code.

Algorithm 3: EHO-FWGWO Algorithm

Input ← grey wolf population X, number of iteration Tmax, size N, swing factor C, control parameter a, and learning factors c1, c2

Output ← Optimal solution (α)

1. Begin
 2. **For** $i = 1:N$
 3. **For** $j = 1:d$
 4. Generate Tent Chaotic sequence according to Eq. (19)
 5. **End for**
-

(Continued)

Algorithm 3: (Continued)

```

6.      End for
7.      // use tent to initialize the FWGWO
8.      For  $t = 1: Tmax$ 
9.      Update a according to Eq. (9)
10.     For  $i = 1:N$ 
11.     For  $j = 1: d$ 
12.     Calculate  $\vec{D}_\alpha, \vec{D}_\beta, \vec{D}_\gamma$  according to Eqs. (16)–(18)
13.     Calculate  $\vec{X}_1, \vec{X}_2$  &  $\vec{X}_3$  according to Eqs. (11)–(13), (21)
14.     Update position using Eq. (21)
15.     End for
16.     End for
17.     Calculate fitness value and then update  $\alpha, \beta,$  and  $\delta$ 
18.     End for
19.     End

```

3.1.2 Embedded Algorithm

This work embeds SVM-t for statistically analyzing features from a dataset for selection as SVMs create a maximal hyper plane from feature vectors to find classes in a dataset. The proposed SVM-t takes advantage of the most significant feature subsets in a dataset using Eq. (27). Two-sample t-statistics in two directions evaluate variances in the determined classes and thus minimizes variations between them for classifying data points into classes.

$$|t_j| = \left| \frac{(u_j^+ - u_j^-)}{\sqrt{((s_j^+)^2/n^+) + ((s_j^-)^2/n^-)}} \right| \quad (27)$$

where, n^+ (resp., n^-)—number of support vectors designed for class + 1 (resp., -1). Compute mean u_j^+ (resp., u_j^-) and standard deviation s_j^+ (resp., s_j^-) by using simply the support vectors of feature j -labelled class + 1 (resp., -1) to calculate the score of every one attribute. The computed features with improved scores represent features that vary in classification. Thus, SVM-t's perceptive choice of features results in an accurate feature set.

3.2 Aggregation Strategy

Fig. 5 depicts the overall framework of HEFSM in feature selections. HEFSM uses filters, wrappers and embedding for selecting significant features and then orders them based on their importance to produce many sorted optimal feature subsets namely $FS_1, FS_2 \dots FS_t$. Each feature j in FS_i is normalized by using $(n-j)/n$ where n is the total features for computing the feature weight set of the i^{th} feature selection $Wg_i = \{wg_1^i, wg_2^i \dots wg_n^i\}$. The proposed method uses geometric means to compute the total weight of each feature in the set $FS_1, FS_2 \dots FS_t$, and then sorts features based on the total weight for getting the sorted feature sequence Wg . Geometric mean used in this work is defined as the n^{th} root (n -count of numbers) of the product of the features.

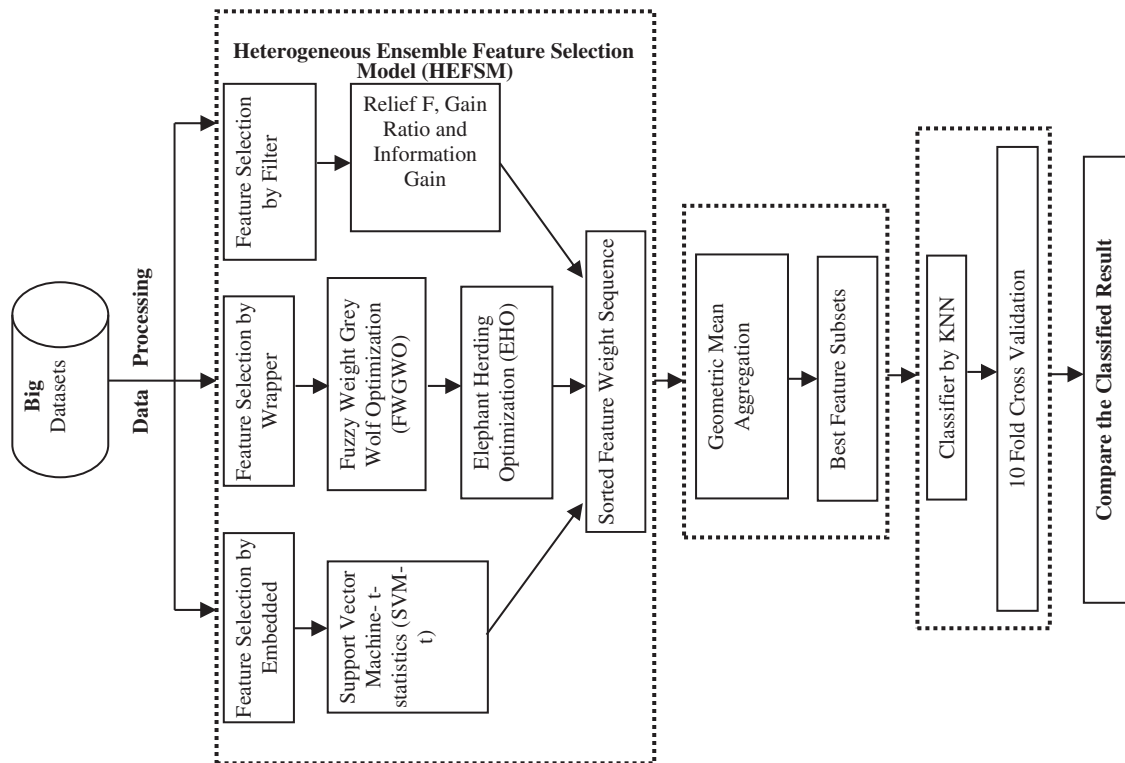


Figure 5: Overall framework of HEFSM method

Geometric mean is obtained by dividing the sum of logarithmic values by the number of features as shown in Eq. (28),

$$\text{geometric mean} = \left(\prod_{i=1}^n (FS_i)^{(1/n)} \right) \quad (28)$$

A threshold percentage $th\%$ selects features from this sorted sequences to create an optimal feature subset. All feature selection methods results are joined for forming a single input for the KNN classifier which then validates the proposed scheme's selected feature sets.

3.3 KNN Classifier

KNN uses Euclidean distances for finding nearest neighbours to data points [30] using Eq. (29).

$$\text{Euc}_d(P, Q) = \sqrt{\sum_{i=1}^d (Q_i - P_i)^2} \quad (29)$$

where, Q_i and P_i -specific attributes in a sample, $i = \{1..d\}$, d – count of used features. One part of the dataset is used for validations while the balance is used for classifications. Since classifiers overfit values, Cross-validation is used to overcome the problem. This work uses a K-fold cross-validation where $K = 10$ assuming samples are divided into K partitions with similar size in cross validations. KNN is trained with $K-1$ partitions for predicting labels or classes in the dataset. These classifications are matched against wrong labels for obtaining the classification error percentage. The results obtained from multiple data rounds have been proven to be statistically reliable [31].

4 Results and Discussion

This section displays experimental results of the proposed HEFSM is implemented on MATLAB (2014a) and on an intel core i7/16Gb running 64 bit windows 10. The proposed scheme was examined using known datasets (listed in Tab. 1) with feature selection issues for its ability to select optimal feature sets from datasets. These datasets were obtained from the UCI Machine Learning Repository website [32]–[34]. Breast cancer Wisconsin data collection, characteristics are determined from a digitized representation of a fine needle aspirate (FNA) of a breast mass. They explain the existence of the cell nuclei features that are in the picture. It contains information on the source, donor, clinical studies, id number, diagnosis (malignant = M, Benign = B) and distribution of grades. Ten real-valued functions are determined for each cell nucleus as pursues: radius, texture, perimeter, area, smoothness, compactness, concavity, concave points, symmetry and fractal dimension (See Tab. 1).

Table 1: Dataset description

No.	Dataset	Instances	No. of features	Classes
1	KC1	2110	21	2
2	WDBC	569	31	2
3	SCENE	2407	299	2
4	SEGMENT	2310	20	7

All the characteristics are measured for the mean, regular error, and “worst” or highest (mean of the three highest values). 80% of the training data and 20% of the evaluation data are included in the categorization phase for the assigned series of data. Data source of Chronic Kidney Disease (CKD) consist of 400 occurrences with 25 characteristics. This dataset is also used to assess CKD and can be obtained approximately within 8 weeks of time from the hospital (Tab. 2). The scene dataset is an image classification task where labels like Beach, Mountain, Field, and Urban are assigned to each image. Segment instances were drawn randomly from a database of 7 outdoor images. The images were hand-segmented to create a classification for every pixel. Each instance is a 3×3 region.

Table 2: Breast cancer wisconsin dataset

S.no	Features name	Type
1	ID number	Numerical
2	Radius(mean of distances from center to points on the perimeter)	
	2.1. Mean radius	Digits
	2.2. Radius SE	Digits
	2.3. Worst radius	Digits
3	Texture (standard deviation of gray-scale values)	
	3.1. Mean texture	Digits
	3.2. Texture SE	Digits
	3.3. Worst texture	Digits

(Continued)

Table 2 (continued)

S.no	Features name	Type
4	Perimeter	
	4.1. Mean perimeter	Digits
	4.2. Perimeter SE	Digits
	4.3. Worst perimeter	Digits
5	Area	
	5.1. Mean area	Digits
	5.2. Area SE	Digits
	5.3. Worst area	Digits
6	Smoothness (local variation in radius lengths)	
	6.1. Mean smoothness	Digits
	6.2. Smoothness SE	Digits
	6.3. Worst smoothness	Digits
7	Compactness (perimeter²/area-1.0)	
	7.1. Mean compactness	Digits
	7.2. Compactness SE	Digits
	7.3. Worst compactness	Digits
8	Concavity (severity of concave portions of the contour)	
	8.1. Mean concavity	Digits
	8.2. Concavity SE	Digits
	8.3. Worst concavity	Digits
9	Symmetry	
	9.1. Mean symmetry	Digits
	9.2. Symmetry SE	Digits
	9.3. Worst symmetry	Digits
10	Fractal dimension(“coastline approximation”-1)	
	10.1. Mean fractal dimension	Digits
	10.2. Fractal dimension SE	Digits
	10.3. Worst fractal dimension	Digits
11	Diagnosis(M = Malignant, B = Benign)	Categorical
12	Missing attribute	No
13	Class distribution	Categorical
	1. Benign- 357	
	2. Malignant- 212	
	3. Total 569 instances	

Note: Exactness for a category is the quantity of real positives (i.e., the sum of objects appropriately marked as referring to the positive category) divided by the overall quantity of things marked as referring to the positive category (i.e., the aggregate of true positives and false positives, which are objects falsely marked as belonging to the class). It is defined in [Eq. \(30\)](#),

$$Precision = \frac{\text{True Positive(TP)}}{\text{True Positive(TP)} + \text{False Positive(FP)}} \quad (30)$$

In this case, recall is specified as the quantity of true positives divided by the overall elements truly belonging to the positive class. It is defined in Eq. (31)

$$Recall = \frac{\text{True Positive(TP)}}{\text{True Positive(TP)} + \text{False Negative(FN)}} \quad (31)$$

F-measure is defined as combination of precision and recall. It is defined in Eq. (32)

$$F - \text{Measure} = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \quad (32)$$

Classification accuracy is the ratio of correct predictions to total predictions made. It is often described in Eq. (33),

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \quad (33)$$

The classification accuracy of the KNN classifier conducted in Tab. 3. The resulted accuracies in most datasets approved the superiority of the proposed model. In the scene and segment dataset, the HEFSM achieved better results of 92.85% and 97.23% which is better than other methods. The results in Tab. 3 graphically modelled in the following Figs 6–10. Combining experiments were performed using four feature selection methods and four datasets, shown as 3.

Table 3: Results comparison vs. methods

Dataset	Methods	Precision (%)	Recall (%)	F-measure (%)	Accuracy (%)	Computation Time (Seconds)
WDBC	CWOA	90.00	80.11	85.00	91.00	9.2
	GWO	93.99	82.01	87.59	94.00	8.4
	PSO-GWO	96.99	85.00	90.61	97.00	7.5
	HEFSM	97.99	86.02	91.61	98.00	6.9
KC1	CWOA	78.00	78.00	77.00	78.00	12.15
	GWO	79.47	80.14	79.80	80.84	11.86
	PSO-GWO	81.43	82.20	81.82	82.84	8.87
	HEFSM	83.62	84.51	84.06	84.83	7.63
SCENE	CWOA	88.12	89.00	85.00	88.56	92
	GWO	90.81	90.87	86.80	90.82	86
	PSO-GWO	84.56	91.66	87.97	91.82	65
	HEFSM	86.16	92.75	89.33	92.85	52
SEGMENT	CWOA	91.00	90.00	91.00	91.00	15.82
	GWO	94.33	93.75	93.75	94.28	13.93
	PSO-GWO	96.25	95.14	95.69	96.24	10.75
	HEFSM	97.24	96.23	96.73	97.23	9.82

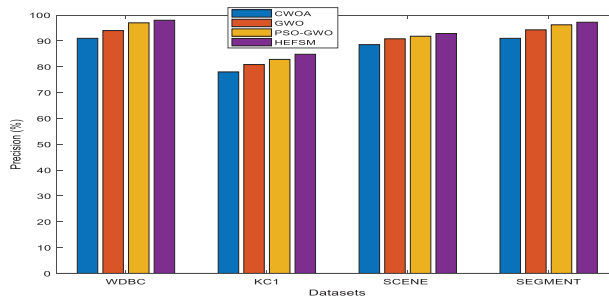


Figure 6: Precision comparison of feature selection methods vs. datasets

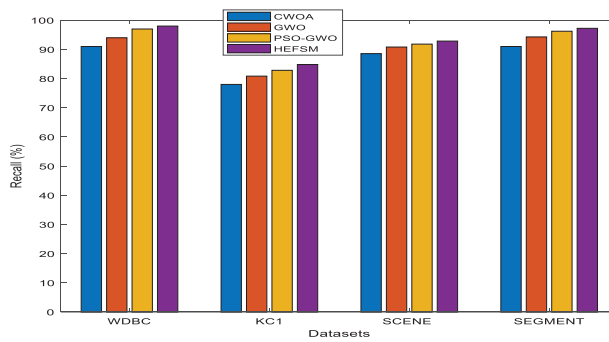


Figure 7: Recall comparison of feature selection methods vs. datasets

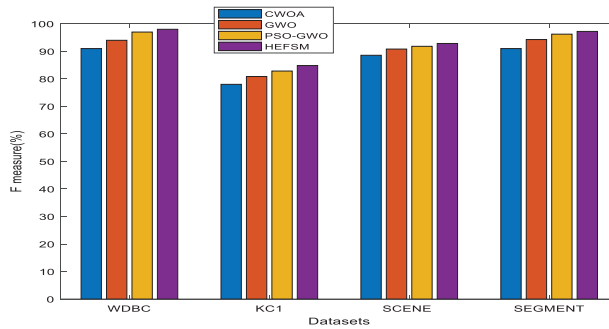


Figure 8: F-Measure comparison of feature selection methods vs. datasets

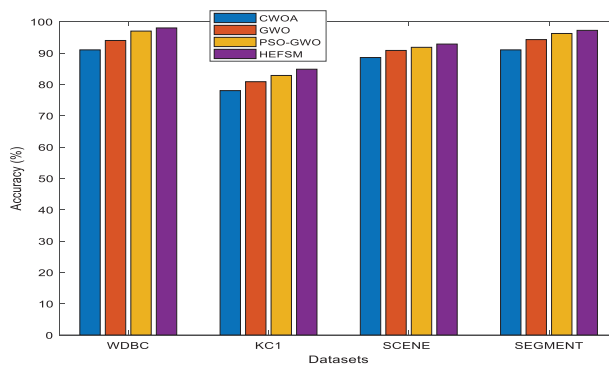


Figure 9: Accuracy comparison of feature selection methods vs. datasets

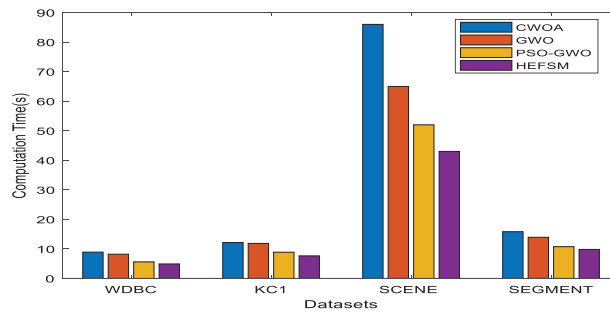


Figure 10: Computational time comparison of feature selection methods vs. datasets

Fig. 6 shows the precision comparison among the proposed HEFSM system and other algorithms concerning the KNN classifier. Fig. 6, the precision result of proposed HEFSM system gives higher results of 97.99% whereas other methods such as CWOA, GWO and PSO-GWO gives lesser precision results of 90.00%, 93.99%, and 96.99% respectively at WDBC dataset. The recall results of feature selection methods are graphically explained in Fig. 7. The proposed model achieved higher recall results of 86.02%, whereas other methods such as CWOA, GWO and PSO-GWO algorithm gives recall result of 80.11%, 82.01% and 85.00% respectively at WDBC dataset concerning the KNN classifier. F-measure results of proposed system and existing feature selection algorithms via KNN classifier are graphically shown in Fig. 8. The proposed model gives improved f-measure results of 91.61%, whereas other methods such as CWOA, GWO and PSO-GWO algorithm gives f-measure result of 85.00%, 87.59% and 90.61% respectively at WDBC dataset concerning the KNN classifier. From the total number of features in all datasets, the proposed HEFSM model via three feature selection methods gives best features which give improved accuracy results. Fig. 9 shows a comparison among the proposed model and other algorithms concerning the accuracy. Tab. 3 shows the accuracy for experiment computations. The results graphically explained in Fig. 9. The proposed model achieved higher accuracy. The accuracy for all datasets has increased when compared to other feature selection methods respectively. Accuracy results of proposed system and existing feature selection algorithms via KNN classifier are graphically shown in Fig. 9. The proposed model gives improved accuracy results of 98.00%, whereas other methods such as CWOA, GWO and PSO-GWO algorithm gives accuracy result of 91.00%, 94.00% and 97.00% respectively at WDBC dataset via KNN classifier. From the total number of features in all datasets, the proposed HEFSM model via three feature selection methods gives best features which give improved accuracy results.

Fig. 10 shows the comparison time among the proposed model and other algorithms concerning the KNN classifier. Tab. 3 shows the elapsed time for experiment computations. The proposed model achieved less computation time of 6.9 s for WDBC dataset. The total time is 9.2 s for CWOA, 8.4 s for GWO and PSO-GWO equals to 7.5 s.

5 Conclusion and Future Work

This paper has proposed and implemented an ensemble learning method HEFSM for selecting optimal feature sets from high dimensional datasets. Multiple techniques and algorithms are used in this proposed work. It uses filters, wrappers and embedding in its architecture. Filtering is based on ReliefF, InfoG and GR. SVM-t is embedded for statistical evaluations and selections. A new binary variant of wrapper functions in feature selection is executed by integrating FWGWO and EHO algorithms for achieving better performances and achieving a better trade-off between predictive performance and stability. The ensemble paradigm has been introduced in the framework for improving robustness of feature selections,

specifically for high-dimensional or minimal sampled datasets or data where extracting stable features is complex and difficult. The proposed scheme's aggregate learning from multiple optimal feature subsets results in higher stability especially while handling high dimensional data in HEFSM. This framework's EHO-FWGW algorithm overcomes local optima issues by its random initializations, use of TCM and features weights generated by a fuzzy function. Moreover, the application of a sigmoid function in converting the search space to a binary form increases the proposed systems ability. HEFSM technique could find best feature subsets for maximizing classifier accuracy as proved by the results of KNN classifications on the selected optimal feature subsets. For instance, the proposed model gives improved accuracy results of 98.00%, whereas other methods such as CWOA, GWO and PSO-GWO algorithm gives accuracy result of 91.00%, 94.00% and 97.00% respectively at WDBC dataset via KNN classifier. Invest a significant amount of time in data to increase the model's reliability and performance. KNN classifier, accuracy depends on the quality of the data and for larger dataset the prediction stage might be slow. In future some other classifiers such Support Vector Machine (SVM), deep learning methods and fuzzy methods have been developed for classification which is left as scope of the future work.

Funding Statement: The authors received no specific funding for this study.

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

References

- [1] K. Hi'ovská and P. Koncz, "Application of artificial intelligence and data mining techniques to financial markets," *Economic Studies & Analyses/Acta VSFS*, vol. 6, no. 1, pp. 1–15, 2012.
- [2] M. Qiu, S. Y. Kung and Q. Yang, "Secure sustainable green smart computing," in *IEEE Transactions on Sustainable Computing*, vol. 4, no. 2, pp. 142–144, 2019.
- [3] B. Johnson, *Algorithmic Trading & DMA: An introduction to direct access trading strategies*, 1st ed., London: Myeloma Press, pp. 574, 2011.
- [4] J. Cai, J. Luo, S. Wang and S. Yang, "Feature selection in machine learning: A new perspective," *Neuro Computing*, vol. 3, pp. 70–79, 2018.
- [5] E. Emary and H. M. Zawbaa, "Feature selection via levy antlion optimization," *Pattern Analysis and Applications*, vol. 22, no. 3, pp. 857–876, 2018.
- [6] R. C. Chen, C. Dewi, S. W. Huang and R. E. Caraka, "Selecting critical features for data classification based on machine learning methods," *Journal of Big Data*, vol. 7, no. 1, pp. 1–26, 2020.
- [7] J. C. Ang, A. Mirzal, H. Haron and H. N. A. Hamed, "Supervised, unsupervised, and semi-supervised feature selection: A review on gene selection," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 13, no. 5, pp. 971–989, 2015.
- [8] P. Smialowski, D. Frishman and S. Kramer, "Pitfalls of supervised feature selection," *Bioinformatics*, vol. 26, no. 3, pp. 440–443, 2010.
- [9] S. H. Huang, "Supervised feature selection: A tutorial," *Artificial Intelligence Research*, vol. 4, no. 2, pp. 22–37, 2015.
- [10] N. Hoque, M. Singh and D. K. Bhattacharyya, "EFS-MI: An ensemble feature selection method for classification," *Complex & Intelligent Systems*, vol. 4, no. 2, pp. 105–118, 2018.
- [11] M. Sharawi, H. M. Zawbaa and E. Emary, "Feature selection approach based on whale optimization algorithm," in *2017 Ninth Int. Conf. on Advanced Computational Intelligence (ICACI)*, Doha, Qatar, pp. 163–168, 2017.
- [12] G. I. Sayed, A. Darwish and A. E. Hassanien, "A new chaotic whale optimization algorithm for features selection," *Journal of Classification*, vol. 35, no. 2, pp. 300–344, 2018.
- [13] K. Chen, F. Y. Zhou and X. F. Yuan, "Hybrid particle swarm optimization with spiral-shaped mechanism for feature selection," *Expert Systems with Applications*, vol. 128, pp. 140–156, 2019.

- [14] S. Gu, R. Cheng and Y. Jin, "Feature selection for high-dimensional classification using a competitive swarm optimizer," *Soft Computing*, vol. 22, no. 3, pp. 811–822, 2018.
- [15] Q. Tu, X. Chen and X. Liu, "Hierarchy strengthened grey wolf optimizer for numerical optimization and feature selection," *IEEE Access*, vol. 7, pp. 78012–78028, 2019.
- [16] F. Haiz, A. Swain, N. Patel and C. Naik, "A Two-dimensional (2-D) learning framework for particle swarm based feature selection," *Pattern Reorganization*, vol. 76, pp. 416–433, 2018.
- [17] H. M. Zawbaa, E. Emary, C. Grosan and V. Snasel, "Large-dimensionality small-instance set feature selection: A hybrid bio-inspired heuristic approach," *Swarm Evolution Computer*, vol. 42, pp. 29–42, 2018.
- [18] S. Fong, R. Wong and A. Vasilakos, "Accelerated PSO swarm search feature selection for data stream mining big data," *IEEE Transaction. Services Computer*, vol. 9, no. 1, pp. 33–45, 2016.
- [19] M. Abdel-Basset, D. El-Shahat, I. El-henawy, V. H. C. de Albuquerque and S. Mirjalili, "A new fusion of grey wolf optimizer algorithm with a two-phase mutation for feature selection," *Expert Systems with Applications*, vol. 139, pp. 1–14, 2020.
- [20] Q. Li, H. Chen, H. Huang, X. Zhao, Z. Cai *et al.*, "An enhanced grey wolf optimization based feature selection wrapped kernel extreme learning machine for medical diagnosis," *Computer Mathematics Methods*, vol. 2017, pp. 1–15, 2017.
- [21] I. M. El-Hasnony, S. I. Barakat, M. Elhoseny and R. R. Mostafa, "Improved feature selection model for big data analytics," *IEEE Access*, vol. 8, pp. 66989–67004, 2018.
- [22] J. Wang, J. Xu, C. Zhao, Y. Peng and H. Wang, "An ensemble feature selection method for high-dimensional data based on sort aggregation," *Systems Science & Control Engineering*, vol. 7, no. 2, pp. 32–39, 2019.
- [23] B. Pes, "Ensemble feature selection for high-dimensional data: A stability analysis across multiple domains," *Neural Computing and Applications*, vol. 32, pp. 1–23, 2019.
- [24] I. H. Witten, E. Frank, M. A. Hall and C. J. Pal, *Data Mining: Practical Machine Learning Tools and Techniques*, Burlington: Morgan Kaufmann, vol. 2, pp. 1–664, 2016.
- [25] I. Kavakiotis, O. Tsave, A. Salifoglou, N. Maglaveras, I. Vlahavas *et al.*, "Machine learning and data mining methods in diabetes research," *Computational and Structural Biotechnology Journal*, vol. 15, pp. 104–116, 2017.
- [26] H. Dağ, K. E. Sayin, I. Yenidoğan, S. Albayrak and C. Acar, "Comparison of feature selection algorithms for medical data," in *2012 Int. Symp. on Innovations in Intelligent Systems and Applications*, Karadeniz Technical University, pp. 1–5, 2012.
- [27] S. Mirjalili, S. M. Mirjalili and A. Lewis, "Grey wolf optimizer," *Advances in Engineering Software*, vol. 69, pp. 46–61, 2014.
- [28] Z. J. Teng, J. L. Jv and L. W. Guo, "An improved hybrid grey wolf optimization algorithm," *Soft Computing*, vol. 23, no. 15, pp. 6617–6631, 2019.
- [29] G. G. Wang, S. Deb and L. D. S. Coelho, "Elephant herding optimization," in *2015 3rd Int. Symp. on Computational and Business Intelligence (ISCBI)*, Indonesia, pp. 1–5, 2015.
- [30] J. Gou, H. Ma, W. Ou, S. Zeng, Y. Rao *et al.* "A generalized mean distance-based k-nearest neighbor classifier," *Expert System Applied*, vol. 115, pp. 356–372, 2019.
- [31] T. T. Wong and P. Y. Yeh, "Reliable accuracy estimates from k-fold cross validation," *IEEE Transtions Knowledge Data Engineerng, Early Access*, vol. 2, pp. 635–651, 2019.
- [32] E. D. Dheeru and K. Taniskidou, "UCI machine learning repository," *Technical Representation*, vol. 5, pp. 586–592, 2017.
- [33] W. Sun, X. Chen, X. R. Zhang, G. Z. Dai, P. S. Chang *et al.*, "A Multi-feature learning model with enhanced local attention for vehicle re-identification," *Computers, Materials & Continuous*, vol. 69, no. 3, pp. 3549–3560, 2021.
- [34] W. Sun, G. C. Zhang, X. R. Zhang, X. Zhang and N. N. Ge, "Fine-grained vehicle type classification using lightweight convolutional neural network with feature optimization and joint learning strategy," *Multimedia Tools and Applications*, vol. 80, no. 20, pp. 30803–30816, 2021.