Computer
& Robotics

A Review of Feature Selection Method Based on Optimization Algorithms

Zohre Sadeghian[a], Ebrahim Akbar[a,]*, Hossein Nematzadeh[a], Homayoun Motameni[a]

[a]Department of computer engineering, Sari Branch, Islamic Azad University, Sari, Iran
Received 10 December 2022;Accepted 23 December 2022

Abstract

Feature selection is the process of identifying relevant features and removing irrelevant and repetitive ones to establish a subset of features describing the problem well and with minimal loss of efficiency. One of the feature selection approaches is the use of optimization algorithms. This work provides a summary of some meta-heuristic feature selection methods proposed from 2018 to 2021, which have been designed and implemented on a wide range of data. The results of the study showed that some meta-heuristic algorithms alone cannot perfectly solve the feature selection problem on all types of datasets with an acceptable speed. In other words, depending on the available dataset, a suitable meta-heuristic algorithm should be used.

Keywords: Data dimension reduction; Classification; Feature selection; Optimization algorithms; Meta-heuristic algorithms

1. Introduction

In recent decades, with the progress of data collection/storage technologies and the growing mass of high-dimensional data in various scientific fields, particularly data mining, data dimension reduction has become a fundamental issue. Methods proposed in this regard are generally divided into two categories [1-4]:

1) Feature Extraction methods that map a multidimensional space to a smaller space by combining the values of existing features. The obtained features contain all or most of the information contained in the original features [5-12].
2) Feature Selection methods that attempt to reduce the size of data by selecting a subset of original features [13-19].

Feature selection is often preferred in many fields since it preserves the physical perception of the original features by keeping some important features and provides better readability and interpretability of the models [16, 20]. One of the feature selection approaches is the use of optimization algorithms that have been widely studied in recent years and have been found largely successful [15, 21]. Several papers have reviewed the optimization-based feature selection methods. The authors in [22] studied the Particle Swarm Optimization (PSO)-based feature selection methods proposed before 2010. In [23], the researchers investigated Evolutionary Computation (EC)-based feature selection methods. However, their study was limited to the methods designed based on Genetic Algorithm (GA), Genetic programming (GP), PSO, and Ant Colony Optimization (ACO), which have been published before 2015. The authors in [24] reviewed the Swarm Intelligence (SI)-based feature selection methods proposed from 2001 to 2017. They classified the methods based on the initialization and search mechanism. In another paper [25], SI-based methods were categorized based on the representation and the search mechanism. These methods included PSO, Artificial Bee Colony (ABC), and ACO-based feature selection methods presented in the literature before 2018.

The papers mentioned above are limited to the years before 2019 and some specific algorithms or domains. In addition, criteria such as fitness function and classifiers used in some studies have not been investigated. Accordingly, this study reviews and

* Corresponding Author. Email: ebrahimakbari30@yahoo.com

compares optimization-based feature selection methods proposed from 2018 to 2021. These methods are based on 10 different optimization algorithms. The major contributions of this article can be summarized as:

- Introducing optimization-based feature selection methods from 2018 to 2021
- Reviewing the performance evaluation criteria of the algorithms proposed in the literature
- Reviewing and comparing the fitness functions used in these methods

The remainder of this paper is organized as follows: Section 2 describes the basic concept of feature selection. Next, Section 3 reviews the studies published from 2018 to 2021 based on optimization algorithms. Then, Section 4 provides an analysis of these methods. Finally, the research conclusion and future trends in feature selection are presented in Section 5.

2. Feature Selection

Feature selection is the process of finding a minimal subset of features, which contains the necessary and sufficient information for the intended purpose. Different feature selection methods introduced in the literature attempt to find the best subset; however, creating an optimal subset of features out of medium- and large-sized datasets is difficult and highly expensive. The feature selection process involves the following four basic steps: Subset generation, Subset evaluation, Stopping criterion, and Result validation [26] (see Fig 1).

Depending on the feature selection mechanism, feature selection methods can be classified into three categories: filter, wrapper, and embedded [27]. Filter-based methods calculate a rank or a score for each feature using data dependency techniques, and then remove the features with lower scores. The advantages of these methods are the low computational cost, acceptable generalizability, high speed, and applicability to high-dimensional data. However, they may not be suitable to target learning algorithms due to the lack of a learning algorithm in search phase. The wrapper methods include a learning algorithm as a black box and use its predictive performance to assess the selected features. These methods involve two steps: searching for a features subset and evaluating the subset. The disadvantages of these methods are high search space for high-dimensional data, high computational complexity, low speed, and being time consuming. Finally, the embedded methods use the filter-based methods to reduce data size in the first step, and then apply the wrapper methods to select the best feature subset in the second step. These methods remove redundant and irrelevant features without significantly decreasing the speed or increasing the computational complexity. The embedded methods perform better than the wrapper ones because they do not need re-evaluation of feature subsets. Most meta-heuristic algorithms are considered wrapper methods because they generate a subset of solutions during a given iteration, then, evaluate them at each iteration, and finally, extract the best solution.

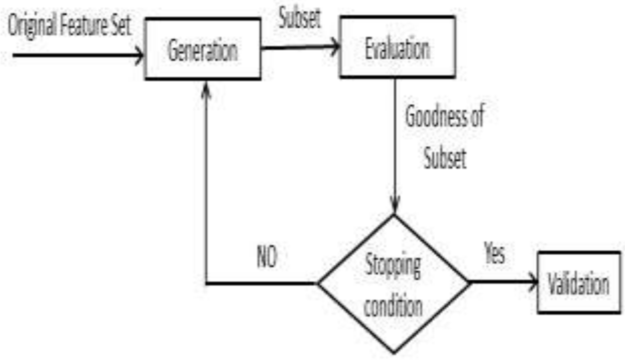

Fig. 1. Feature selection process

3. Meta-Heuristic Algorithms for Feature Selection

Nowadays, with the rapid growth of real-world problems and the importance of quick access to answers, the use of optimization algorithms has grown significantly. Unlike classical methods, optimization search methods perform space searching in parallel and use only one fitness function to guide the search. They are able to discover the answer due to their swarm intelligence [28]. In this section, the optimization-based feature selection methods published from 2018 to 2021 are reviewed.

3.1. Genetic Algorithm-Based Feature Selection

Genetic Algorithm (GA) was introduced by [29] and was developed by [30]. This algorithm uses two operators, i.e., mutation and crossover, for survival of the best and mating processes. The structural diversity of a population is increased by the mutation operator. From this point of view, a mutation operator is often known as a heuristic operator. From another point of view, it can be thought as an exploitation operator due to the conservation of genetic material. On the other hand, the crossover operator produces better offspring by combining two or more parents. In this view, a crossover operator is also recognized as an exploitation operator. However, an acceptable crossover operator should generate individuals in the exploration zone. Therefore, a crossover operator cannot be considered an exploration operator, while the mutation operator is a pure exploitation operator. In recent years, many researchers have used GA to solve the problem of feature selection, e.g., GIFS [14], GA-enhanced PLSR [31]. Table 1. demonstrates the primary details extracted from the chosen recently-published GA-based papers.

Table 1
Details of methodology and findings of the GA-based feature selection algorithms

Key NO.	Reference	Methodology / Fitness function / Stop condition / Finding (Advantages or Disadvantage)
1	[32]	It uses the Improved Binary GA with Feature Granulation (IBGAFG) at the first phase to select important feature, Improved Neighborhood Rough Set with sample Granulation (INRSG) at the second phase to select best feature subset, and Granularity Optimization-based GA (ROGA) to obtain the optimal granularity parameters.
		Classification accuracy
		500 iterations
		1) Obtaining granular parameters in a self-adaptive manner 2) Proving the applicability of IBGAFG to large-scale data 3) Being limited in the field of pattern recognition and bioinformatics
2	[33]	It combines genetic operations (global search) and Hybrid $L_{1/2} + L_2$ Regularization (HLR) embedded method (local search).
		Classification accuracy
		Reaching a specific number of features
		1) Selecting effectively the relevant features, predicting the patients' class, and constructing the accurate learning model in high-dimensional biological datasets. 2) Using a practical tool for learning prediction
3	[34]	It employs the Support Vector Machine (SVM) and GA to select more significant features.
		Classification accuracy
		Max iterations
		Good performance with SVM classifier compared to Multilayer Perceptron (MLP), K Nearest Neighbor (KNN), and J48 classifiers
4	[35]	It uses the Great Deluge Algorithm (GDA) [36] instead of mutation operations in GA.
		Classification accuracy
		20 iterations
		1) GA uses DA's local searching strategy to move around the local optimum and reach the global optimum. 2) GDA enhances exploitability of GA.
5	[37]	It utilizes GA and Elastic Net (EN)
		$\alpha\, r_{RMSE} + (1-\alpha) w_p$ |selected predictor |/|total predictors| , where r_{RMSE} is defined based on the average root mean square error and the average of response variable.
		The maximum number of generations or the lack of fitness improvement in two consecutive generations
		1) Reducing the computation time of finding the best subset 2) Reducing the probability of redundant/irrelevant predictors by using EN

3.2. Ant Colony Optimization-Based Feature Selection

Ant Colony Optimization (ACO) pioneered in [38] is inspired by the feeding behavior of ants to find the shortest path. This algorithm can choose solutions for the problem by simulating two actions: pheromone spraying and evaporation; these solutions can gradually approach the global optimal solution. The exploitation rate is higher than exploration in this algorithm; as a result, it has a high convergence speed. In recent years, many researchers have proposed methods based on this algorithm (e.g., MRMR Enhanced ACO (MRMR-EACO) [39], Modified Binary coded ACO (MBACO) [40] to solve the feature selection problem. Table 2 gives the primary details of the chosen recent ACO-based papers.

3.3. Grasshopper Optimization Algorithm-Based Feature Selection

Grasshopper Optimization Algorithm (GOA) was first proposed in [41]. This algorithm mimics the swarming behavior of grasshoppers in two phases: exploration (random movements) and exploitation (local movements). The repulsive force of grasshoppers causes them to move away from each other and explore the search space extensively. This is the main reason for high exploration and local optima avoidance of GOA. The attractive forces between grasshoppers quickly drive them to the best solution achieved so far, which enables exploitation. Moreover, GOA uses a comfort zone coefficient to make a balance between exploration and exploitation. Table 3 presents the primary details of the recent GOA-based feature selection methods.

Table 1
Details of methodology and findings of the ACO-based feature selection algorithms

Key NO.	Reference	Methodology / Fitness function / Stop condition / Finding (Advantages or Disadvantage)
6	[42]	It employs ACO and feature selection method.
		$\alpha FPR + (1-\alpha)\, N_{selected}/N_{total}$, where FPR represents the false positive rate of classification and is calculated by $FP/(TN + FP)$. $\alpha = 0.7$
		40 iterations
		1) Increasing search speed by the two-stage pheromone updating 2) Doing better search in the feature space and achieving good execution time
7	[43]	It uses a graph to analyze the dependence between features, the Fisher Score (F-Score) to analyze the relevance of features, the absolute of Pearson's correlation to analyze the redundancy, and the multiple discriminant analysis (MDA) [44] to update the amount of pheromone and select the k best features based on the pheromone values.
		50 iterations
		MDA [44]
		1) Selecting more relevant features, hence improving the accuracy by using multiplication operator instead of the subtraction and initialization of pheromones 2) Being limited to small and medium datasets
8	[45]	It applies the Text Feature Selection ACO (TFSACO) (Wrapper method) and UFSACO (Filter method)
		$\alpha\ Accuracy(S) + \beta\ (\lvert N_{unselected}\rvert)/N_{total}$, where $\alpha = 1$ and $\beta = 100$
		20 iterations
		1) Avoiding premature convergence and exploring search space better by restricting pheromone values to [0-1] 2) Increasing effectiveness and accuracy using fitness-based memory 3) Being limited to small and medium datasets
9	[46]	It uses ACO to evaluate the selection process and ANN to find the best subset
		The ANN classification accuracy
		The best subset with the least classification error
		1) Offering a high effectiveness on big data. 2) Being limited to textual data sets

Table 2
Details of methodology and findings of the GOA-based feature selection algorithms

Key NO.	Reference	Methodology	Fitness function	Stop condition	Finding (Advantages or Disadvantage)
10	[47]	It employs GOA, selection operators, and Evolutionary population Dynamics (EPD) [48].	$\alpha\ (1 - Accuracy)\ + (1 - \alpha)\ N_{selected}/N_{total}$, where $\alpha = 0.99$.	100 iterations	1) Great effect EPD on GOA performance. 2) Improving the convergence of selection operators and increasing the ability of finding the best solution 3) Higher average CPU time compared to GA, PSO, and Binary GWO (BGWO)
11	[49]	It employs S-shaped [50] and V-shaped [51] transfer functions to convert GOA's solutions to binary.	Like key 10	100 iterations	1) Improving exploration and increasing the performance of BGOA by the mutation operator 2) Being limited to certain data
12	[52]	It uses the Hamming distance to normalize the distance between grasshoppers and NB to evaluate feature subset.	Like key 10	100 iterations	1) Improving the exploration and computational time by the binary initialization of population 2) Showing better performance compared to Binary Dragonfly Optimization (BDO), BGOA, BGWOA, and Naïve Bayes PSO (NBPSO) on large datasets 3) Being limited to certain data

Table 3. (Continued).
Details of methodology and findings of the GOA-based feature selection algorithms

Key NO.	Reference	Methodology	Fitness function	Stop condition	Finding (Advantages or Disadvantage)
13	[53]	It applies improved SVM and GOA.	The SVM classification accuracy	Maximum iterations	1) Obtaining the highest accuracy by increasing the number of search factors 2) Increasing the convergence speed of this method for large real data sets 3) Being limited to certain data
14	[54]	It employs an adaptive reducing parameter (for shrinking comfort, repulsion, and attraction areas, which, in turn, makes a balance between exploration and exploitation), a probability-based distribution factor (for substituting the duplicate features), and rounding operation (for Out-of-range indices).	The SVM or KNN classification error rate	100 iterations	1) Searching promising features and finding the global solution by the feature goodness factor. 2) Not paying attention to redundancy

3.4. Particle Swarm Optimization-Based Feature Selection

Particle Swarm Optimization (PSO), which was first proposed in [55], was inspired by the swarm movement of birds in search of food. In this algorithm, each solution (particle) has a velocity vector to guide the particle's motion, a position vector to identify the particle's location, and a fitness value to measure the particle's suitability. An important factor to the success of PSO is the balance between exploitation (local search) and exploration (global search). Exploration is typically done at the initial steps of the search, but gradually gives way to exploitation of promising solutions as the search progresses. However, PSO often quickly converges to a local minimum when working with multimodal functions, hence missing better opportunities. To solve this problem, methods such as non-global best neighbourhoods have been proposed, which increase exploration but result in reduced convergence. In recent years, several researchers have used this algorithm to solve the feature selection problem [56, 57]. Table 3 presents the primary details extracted from the recent studies conducted on the PSO-based feature selection.

3.5. Gray Wolf Optimization-based Feature Selection

Gray Wolf Optimization (GWO) was first proposed in [58]; it mimics the social behaviour and hierarchy of gray wolves while hunting. The hunting process of gray wolves has three stages: tracking and approaching (exploration), pursuing and encircling, and attacking (exploitation). Despite its acceptable performance in unknown, challenging search spaces and solving semi-real and real problems, this algorithm has limited exploration ability to solve complex high-dimensional problems and may get stuck in local optima in the middle of execution time. In this case, the diversity of the population gradually decreases and in some cases it is not able to escape from these local optima and reaches premature convergence. In recent years, various GWO-based feature selection methods have been proposed, e.g., bGWO [59], GWO-ANN [60].

Table 4 gives the primary details of some of these Methods.

3.6. Butterfly Optimization Algorithm-Based Feature Selection

The Butterfly Optimization Algorithm (BOA) was first proposed by [83] was inspired by the searching behaviors of butterflies. Each butterfly produces scent that can be sensed by neighboring butterflies, which forms a general system of social learning. Each butterfly moves toward the best butterfly in the search space, which is called global search phase (exploration); when it cannot detect the fragrance network in the search space, it takes random steps, which is called the local search phase (exploitation). BOA has shown acceptable results in terms of discovery, exploitation, and convergence. However, BOA sometimes suffers from reduced population diversity and tendency to get stuck in local optima. gives the primary details extracted from recently-proposed BOA-based feature selection algorithms.

Table 3
Details of methodology and findings of the PSO-based feature selection algorithms

Key NO.	Reference	Methodology / Fitness function / Stop condition / Finding (Advantages or Disadvantage)
15	[61]	1) It divides randomly the population into two groups, carries out pairwise competitions between the particles from each group, passes directly winner particle to the next iteration, and updates the position and velocity of the loser particle by learning from the winner particle; 2) It uses an archive technique to record the fitness values of all previous feature subsets.
		The average error rates of KNN classifier
		200 iterations
		1) Reducing the search time using the archive technique 2) Being limited to certain data
16	[62]	It applies 1) Predictive Gene Pre-Filtering (PGPF) phase, and 2) Gene Optimization and Cancer Classification (GOCC) phase using IBPSO-NB wrapper method and 10-fold cross-validation.
		The NB classification error
		100 iterations
		1) Offering an effective tool for the DNA microarray analysis 2) Showing lower complexity than the FCBF, BPSO, PSO-DT, Markov Blanket-Embedded Genetic Algorithm (MBEGA), and Taguchi Chaotic Binary Particle Swarm Optimization (TCBPSO) methods
17	[63]	It uses iBPSO and then SFLA [64] to obtain the optimal feature subset.
		The NB classification accuracy
		500 iterations
		1) Enhancing the search speed and creating a balance between exploration and exploitation by using Inertia weight in

		the velocity of iBPSO 2) Being limited to textual datasets
18	[65]	It employs Set-Based PSO (SBPSO) [66] and KNN classifier.
		The average of KNN classification accuracy
		Achieving the accuracy of 100% or not improving the best fitness in 50 iterations or passing the maximum number of iterations
		1) Showing better performance of KNN classifier in SBPSO compared to the Gaussian NB and DT J48 2) Being limited to small and medium size datasets
19	[67]	It uses 1) a logistic map sequence [17] to update the inertial weight in particle velocity formula, 2) Two dynamic parameters in position update formula to enhance the quality of position in the next generation, and 3) A spiral-shaped mechanism [68] to enhance the solution quality.
		Like key 10
		100 iterations
		1) Improving the population diversity using logistic map sequencing 2) Having a high computational time complexity
20	[69]	It applies the Modified-BPSO method by using Silhouette index (SI) [70] to select the best swarm.
		SI
		150 iterations
		1) Showing a high accuracy by selecting high SI value 2) Being limited to small and medium size datasets
21	[71]	It uses 1) the Average Mutual Information (AMI)-based space reduction strategy [72] to remove irrelevant and weakly-relevant features, 2) feature redundancy- based local filter search strategy to delete the redundant features and add the missing important features, and 3) similarity-based assessment function and a parameter-free update strategy to get the high performance.
		$fit_s - fit_{dis}$, where fit_{dis}, and fit_s are dissimilarity of the selected features and the similarity of unselected features.
		100 iterations
		1) Removing irrelevant and redundant features quickly and increasing the convergence speed and exploitation 2) Being limited to certain data and small size datasets
22	[73]	It employs an integrative BPSO feature selection and a hybrid PSO-KMeans algorithm
		Like key 10
		15 iterations
		1) Improving the convergence speed and accuracy. 2) Having a Low execution time 3) Being dependent on user-defined parameters

Table 4
Details of methodology and findings of the GWO-based feature selection algorithms

Key NO.	**Reference**	**Methodology**
		Fitness function
		Stop condition
		Finding (Advantages or Disadvantage)
23	[74]	It uses LF [75] to increase the step size of a search agent in GWO.
		The SVM accuracy
		Not mentioned
		1) Creating a good balance between exploration and exploitation 2) Removing irrelevant and redundant features while maintaining high classification accuracy 3) Being limited to certain data
24	[76]	It employs the binary version of PSOGWO [77] and the KNN classifier with the Euclidean separation matrix to find the best solution.
		Like key 10
		100 iterations
		1) Creating a good balance between exploration and exploitation 2) Being limited to binary datasets
25	[78]	Modified version of GWO

		$Accuracy + \alpha(1 - \lvert N_{selected}\rvert / \lvert N_{total}\rvert)$, where Weighted parameter α is set to 0.7
		1000 iterations
		1) Making a balance among the sum of the importance of the selected features, the importance of the candidate feature, and the size of the subset by using the weight factor 2) Being limited to certain data
26	[79]	It uses an enhanced global-best lead strategy to enhance the local search ability of GWO, the adaptable cooperative hunting strategy to increase the population diversity and the ability of global search, and the disperse foraging strategy to make a balance between exploitation and exploration.
		Like key 10
		2500 iterations
		1) Solving effectively real-world optimization problems with high accuracy 2) Being limited to binary datasets
27	[80]	GWO and CSA
		Like key 10
		100 iterations
		1) Solving effectively real-world optimization problems with high accuracy 2) Being superior to GWO, AGWO, and EGWO 2) Being limited to small datasets
28	[81]	It uses the Serial Grey-Whale (HSGW), Random Switching Grey-Whale (RSGW), and Adaptive Switching Grey-Whale Optimization (ASGWO) by the combination of GWO and WOA.
		Like key 10
		100 iterations
		1) Offering better results of HSGW compared to RSGW and ASGW in most of the datasets considered 2) Reporting low computational time for ASGWO
29	[82]	It uses a two-phase mutation to improve the GWO exploitation capability and the Sigmoid and V-shaped functions to transform the continuous search space to binary space
		Like key 10
		30 iterations
		1) Effectively finding the best subset by mutation operator 2) Having high runtime

3.7 Salp Swarm Algorithm-Based Feature Selection

Salp Swarm Algorithm (SSA), which was first proposed in [84], mimics the swarming behavior of salps in search of food. The position of salps is defined based on the number of variables in the given problem. First, salps go around the search space (exploration phase); then, they move towards the global optimum and move locally instead of globally (exploitation phase). SSA uses an adaptive coefficient to create a balance between exploration and exploitation. The results of various experiments showed that SSA can explore the search space efficiently, which avoids a large number of local optima in a search space. In addition, SSA is capable of solving real-world problems with unknown search spaces. **Error! Not a valid bookmark self-reference.** presents the primary details of the chosen SSA-based recent papers.

3.8 Whale Optimization Algorithm-Based Feature Selection

The Whale Optimization Algorithm (WOA) was proposed in [68] based on the encircling and hunting behaviours of humpback whales. WOA has three phases: 1) encircling prey to identify the locations of prey and encircle it, 2) bubble-net attacking (exploitation) based on shrinking encircling and spiral updating position, and 3) searching for a prey (exploration). WOA makes a balance between exploration and exploitation by using the distance control parameter. Despite the acceptable performance and high flexibility of this algorithm, if the optimal member is near the local optimum, the population members will be misled and the algorithm will converge to the local optimum instead of the global optimum.
Table 7:presents the primary details of the chosen papers recently published on WOA- based algorithms.

Table 5.
Details of methodology and findings of the BOA-based feature selection algorithms

Key NO.	Reference	Methodology / Fitness function / Stop condition / Finding (Advantages or Disadvantage)
30	[85]	It uses the Binary BOA (S-bBOA and V-bBOA).
		Like key 10
		100 iterations
		1) Showing the superiority of S-bBOA over the V-bBOA in converge, effectively search, finding the accurate best solution, and enhancing the performance of BOA. 2) Not considering redundancy
31	[86]	It employs 1) the Minimal Redundancy-Maximal New Classification Information (MR-MNCI) [87] to select 20% of the relevant and non-redundant features, and 2) the Information Gain bBOA (IG-bBOA) with a three-purpose fitness function to find an optimized feature subset and, finally, an ensemble similarity-based method
		$\alpha Accuracy + \beta(\|N_{total} - N_{selected}\|/\|N_{total}\|) + \delta Mean(I(X_k;Y))$, where $\alpha = 0.99$, $\beta = 0.001$, and $\delta = 0.009$.
		100 iterations
		1) Offering more diversity to reach the optimal solution with the high accuracy by using the mean of IG 2) Showing an acceptable stability 3) Being applicable to medical datasets

Table 6
Details of methodology and findings of the SSA-based feature selection algorithms

Key NO.	Reference	Methodology / Fitness function / Stop condition / Finding (Advantages or Disadvantage)
32	[88]	It uses two binary SSA-based feature selection methods (S-BSSA and V-BSSA).
		Like key 10
		Not mentioned
		1) Enhancing the performance of methods by promoting exploration 2) Reporting the superiority of S-BSSA over V-BSSA
33	[89]	It employs Binary SSA with synchronous updating rules and TC-based leadership structure.
		Like key 10
		100 iterations
		Improving the accuracy, exploration, and exploitation of SSA by selecting half of the salps as leaders
34	[90]	It combines chaotic maps such as logistic, piecewise, singer, sinusoidal, and tent with SSA
		Like key 10 with $\alpha = 0.9999$
		50 iterations
		Improving the optimal solution and convergence
35	[91]	It combines SSA with PSO.
		Like key 10
		200 iterations
		Enhancing the quality of the SSA in searching and creating diversity in the population by using PSO
36	[92]	It applies the Improved Salp Swarm Algorithm (ISSA) by using the inertia weight to modify the current best solution and KNN classifier to evaluate the solutions.
		Like key 10
		50 iterations
		Enhancing the convergence speed and reliability of SSA by using the inertia weight

Table 7. (Continued).
Details of methodology and findings of the SSA-based feature selection algorithms

37	[93]	It employs Dynamic SSA (DSSA) by using 1) a new formula for slap position updating, which is controlled by the Singer's chaotic map, 2) the Local Search Algorithm (LSA) to modify the current best solution.
		The KNN classification error
		100 iterations
		1) Enhancing the diversity of solutions by new position formula 2) Reducing the computational time and improving the best current solution by using the LSA

Table 7.
Details of methodology and findings of the WOA-based feature selection algorithms

Key NO.	Reference	Methodology / Fitness function / Stop condition / Finding (Advantages or Disadvantage)
38	[94]	It combines the Maximum Pearson Maximum Distance (MPMD) with Improved WOA (IWOA).
		The SVM classification accuracy
		100 iterations
		1) Making more exploration and finding global optimal values by using the voting method in IWOA 2) Being limited to small size datasets
39	[95]	It employs IWOA with two improvements: 1) Elite Opposition-Based Learning (EOBL) at WOA initialization, and 2) Differential Evolution (DE) [96] that involved evolutionary operators, i.e., mutation, crossover, and selection at the end of each iteration
		The SVM classification accuracy
		40 iterations
		1) Enhancing the local search capability of WOA by using DE evolutionary operators 2) Improving the initialization phase of WOA by EBOL
40	[97]	It uses WOA to remove 50% of the irrelevant and less-relevant features, MC to prioritize and sort the remaining features and the majority voting feature selection with threshold 10 on the best feature subsets obtained in the second phase.
		$\sqrt{(x-\mathrm{MeanY})^2-(y-\mathrm{MeanX})^2}$, where MeanX and MeanY are mean value of the first class and second class, respectively.
		100 iterations
		1) Increasing the efficiency of the algorithm to remove irrelevant features by using new fitness function of WOA 2) Identifying properly the interference areas of true and false labels and then selecting the best features by ranking features based on this interference 3) Being limited to binary medical datasets
41	[98]	It applies Binary WOA (bWOA-S) by using the sigmoid (S-shaped) transfer function.
		Like key 10
		Not mentioned
		1) Showing acceptable performance in finding the optimal feature subset 2) Being limited to binary medical datasets

3.9 Firefly Algorithm-Based Feature Selection

Firefly Algorithm (FA) [99] is inspired by the brightness of fireflies in nature to solve engineering and nonlinear multi-quality optimization problems as well as NP-Hard problems. A kind of random search is employed in FA to reach a set of solutions. The two main phases of the algorithm in each iteration are the brightness update phase and the movement phase. Fireflies move towards other fireflies with more Light Intensity in their neighbourhood. In this way, during successive iterations, the collection tends towards a better answer. Despite the advantage of avoiding falling into the local optimum trap, FA suffers from premature convergence and poor global exploration when faced with complex high-dimensional problems. **Error! Not a valid bookmark self-reference.**, the primary details of the selected recent papers published on applying the FA-based algorithms to feature selection problems are presented.

4. Analysis and Discussion

Optimization-based feature selection has been used in many fields such as text and image recognition, computer science, physics, and biology. The data set used by these methods can be divided into three categories: Small (up to 150), Medium (between 151 and 1999), and Large (2000 and more)**.** On the other hand, each feature selection method is evaluated by its classifier, e.g., K-Nearest Neighbor (KNN), Naive Bayes (NB), Support Vector Machine (SVM), Decision tree (DT), Random Forest (RF), Multi-Layer Perceptron (MLP), and other classifiers such as Extreme Learning Machine, Linear Discriminant Analysis, and ZeroR. In addition, as discussed earlier in Section 2, these methods can be placed in one of the filter, wrapper, embedded, or hybrid categories based on their feature selection strategy. Moreover, various criteria are used to evaluate the performance and effectiveness of each method in solving the feature selection problem. In the articles reviewed in this study, the following evaluation criteria were used: accuracy and sensitivity (True positive rate), precision (Positive predictive value), specificity (True negative rate), F-measure (the weighted average between Precision and Recall), average selected subset length, statistical standard deviation, average runtime, Wilcoxon test, Friedman test, Best, Worst and Average Fitness or other criteria such as Acceleration rate, post-hoc test, Iman–Davenport test, and T-test.

Table 9 shows a comparison between the studied methods based on used datasets (in terms of size, field, and the number of classes), techniques, evaluation measures, and classifiers.

Table 8.
Details of methodology and findings of the FA-based feature selection algorithms

Key NO.	Reference	Methodology / Fitness function / Stop condition / Finding (Advantages or Disadvantage)
42	[100]	It uses Random Forest with Binary FA.
		$Accuracy - \alpha \lvert N_{total} \rvert / \lvert N_{selected} \rvert$, where the weight α is varied from 0.05 to 0.5 and its optimal value is fixed at 0.12.
		200 iterations
		1) Effectively reducing the size of selected subset and the computational time by using the penalty-based fitness function 2) Being limited to medical domain and cancer diagnosis
43	[101]	It employs 1) the Logistic chaotic map movements, 2) the SA-enhanced local and global solutions, 3) the diversion of weak solutions by using the mean of swarm leader position and a second best solution, and 4) the best and worst memories strategy to enhance the swarm diversity and move the low-light fireflies toward strong-light fireflies and the weak solutions toward optimal regions.
		$\alpha Accuracy - (1-\alpha)(\lvert S \rvert)^{-1}$, $\lvert S \rvert$ is the size of subset, where $\alpha = 0.9$.
		The maximum number of iterations or finding the optimal solutions
		1) Overcoming premature convergence and reaching the global optima 2) Being limited to small datasets without missing values
44	[102]	It uses Mutual Information-based Firefly Algorithm (MIFA) with C4.5 [103], MIFA wrapper method with Bayesian network [104], and a voting-based feature selection.
		The classification accuracy
		100 iterations
		Improving accuracy compared to MI, MIFA with C4.5, or MIFA with the Bayesian network on the KDD CUP 99 dataset
45	[105]	It uses Min-Max along with z-score normalization [106] to eliminate the noisy data and the Firefly Gravitational ACO (FGACO) method to find the optimal subset.
		$F_{ij}^{d}(t) = G\_\text{force}\ (t)(Mass_i(t)M_{assj}(t))/((Dis_{ij}(t))^n + \varepsilon)$
		Finding the optimal solutions
		1) Having low runtime and cost 2) Solving the convergence and local optima problems 3) Being limited to small size datasets

Table 9.
Comparison between the optimization-based feature selection methods

key	Dataset	Techniques	evaluation measures	classifiers
1	The UCI large datasets in computer science, physics, biology, and life areas	Hybrid	Accuracy and Average Runtime	SVM, KNN, NB, DT
2	The UCI large datasets in biology area	Hybrid	Accuracy, No. Selected feature, Sensitivity, Specificity, and Friedman Test	Other
3	Cleveland heart disease database	Wrapper	Accuracy, No. Selected feature, Sensitivity, and Specificity	SVM, KNN, DT, MLP
4	The UCI datasets in various fields and sizes	Embedded	Accuracy, and No. Selected feature	SVM, KNN, MLP
5	The Maize genetic dataset	Hybrid	Other	Other
6	The KDD CUP99 dataset	Wrapper	Accuracy	SVM, KNN, NB, DT
7	The UCI datasets and sleep EEG data from the Dreams Subjects Database	Wrapper	Accuracy, Sensitivity, No. Selected feature, Specificity, Average Runtime, and Standard Deviation	SVM
8	The UCI datasets in various fields and NIPS2003 FS challenge	Embedded	Accuracy, No. Selected feature, Average Runtime, and Standard Deviation	SVM, KNN, MLP
9	The Reuter's datasets	Hybrid	Accuracy, Sensitivity, No. Selected feature, Precision and F-measure,	Other
10	The UCI datasets in various fields and sizes	Wrapper	Accuracy, No. Selected feature, Sensitivity, Specificity, Standard Deviation, Average Runtime, Wilcoxon Test and Best, Worst and Average Fitness	KNN
11	The UCI datasets in various fields and sizes	Wrapper	Accuracy, No. Selected feature, Standard Deviation, Wilcoxon Test and Best, Worst and Average Fitness	KNN
12	The UCI datasets in various fields and sizes	Wrapper	Accuracy, No. Selected feature, Standard Deviation, Average Runtime, Wilcoxon Test, and Friedman Test	KNN
13	Iraqi cancer datasets and University of California Irvine datasets.	Wrapper	Accuracy and No. Selected feature	SVM
14	The UCI small binary biomedical and life datasets	Wrapper	Accuracy, No. Selected feature, and Standard Deviation	KNN
15	The UCI datasets in various fields and sizes	Wrapper	Accuracy and No. Selected feature	KNN
16	Cancer microarray datasets	Hybrid	Accuracy, No. Selected feature, and Average Runtime	NB
17	Dataset developed [107]	Hybrid	Accuracy and No. Selected feature	SVM, KNN, NB

key	Dataset	Techniques	evaluation measures	classifiers
18	The UCI datasets in various fields and sizes	Wrapper	Accuracy, No. Selected feature, and Friedman Test	KNN
19	The UCI datasets in various fields and sizes	Hybrid	Accuracy, No. Selected feature, Average Runtime, and Wilcoxon Test	KNN
20	The UCI small datasets in various fields	Wrapper	Accuracy, No. Selected feature, Standard Deviation, and other	Other
21	Two biological datasets from the UCI repository, two image datasets, and two text datasets from the ASU repository	Filter	Accuracy, No. Selected feature, and Wilcoxon Test	KNN
22	Genomic datasets	Wrapper	F-measure, Average Runtime, and Other	Other
23	The BOSS base ver. 1.01 dataset	Wrapper	Accuracy, No. Selected feature, Wilcoxon Test, Average Runtime, and Best, Worst and Average Fitness	SVM, KNN, Other
24	The UCI dataset in life field	Hybrid	Accuracy, No. Selected feature, Standard Deviation, Average Runtime, and Best, Worst and Average Fitness	KNN
25	Voice, handwriting (spiral and meander), and speech datasets	Wrapper	Accuracy, No. Selected feature, Sensitivity, and Specificity	KNN, DT, RF
26	The UCI small dataset in computer various fields	Wrapper	Accuracy, No. Selected feature, Standard Deviation, Wilcoxon Test, and Best, Worst and Average Fitness	KNN
27	The UCI datasets in various fields and sizes	Hybrid	Accuracy, No. Selected feature, Standard Deviation, Wilcoxon Test, Friedman Test, and Best, Worst and Average Fitness	KNN
28	The UCI small datasets in various fields	Hybrid	Accuracy, No. Selected feature, Standard Deviation, Average Runtime, and Wilcoxon Test	KNN
29	The UCI datasets in various fields and sizes	Wrapper	Accuracy, No. Selected feature, Standard Deviation, Wilcoxon Test, and Best, Worst and Average Fitness	KNN
30	The UCI datasets in various fields and sizes	Wrapper	Accuracy, No. Selected feature, Standard Deviation, Average Runtime, Wilcoxon Test, Friedman Test, and Best, Worst and Average Fitness	KNN
31	The UCI datasets in various fields and sizes	Hybrid	Accuracy, No. Selected feature, Average Runtime, Sensitivity, Specificity, Precision, F-measure, Standard Deviation, Wilcoxon Test, and Other	SVM, NB, RF
32	The UCI datasets in various fields and sizes	Wrapper	Accuracy, No. Selected feature, Wilcoxon Test, Standard Deviation, Friedman Test, and Best, Worst and Average Fitness	KNN
33	The UCI datasets in various fields and sizes	Wrapper	Accuracy, No. Selected feature, Wilcoxon Test, Standard Deviation, and Average Runtime	KNN
34	The UCI datasets in various fields and sizes	Wrapper	Accuracy, No. Selected feature, Average Runtime, and Best, Worst and Average Fitness	KNN
35	The UCI small datasets in various fields	Hybrid	Accuracy, No. Selected feature, Wilcoxon Test, Standard Deviation, Average Runtime, and F-measure	KNN
36	The UCI datasets in various fields and sizes	Wrapper	Accuracy, No. Selected feature, Average Runtime, and Best, Worst and Average Fitness	KNN
37	The 20 UCI datasets and three Hadith datasets	Embedded	Accuracy, No. Selected feature, Wilcoxon Test, Standard Deviation, Average Runtime, and Best, Worst and Average Fitness	KNN
38	The UCI dataset in various fields and sizes	Hybrid	Accuracy, No. Selected feature, Wilcoxon Test, and Standard Deviation	SVM
39	Four Arabic datasets	Embedded	Accuracy, No. Selected feature, and Best, Worst and Average Fitness	SVM, KNN, NB
40	The UCI binary medical datasets	Embedded	Accuracy, No. Selected feature, Sensitivity, Specificity, and Average Runtime	SVM, DT, NB
41	The UCI small datasets in various fields	Wrapper	Accuracy, No. Selected feature, Standard Deviation, Average Runtime, and Best, Worst and Average Fitness	KNN
42	The UCI small medical datasets.	Wrapper	Accuracy and No. Selected feature	RF
43	The UCI datasets in various fields and sizes	Wrapper	Accuracy, No. Selected feature, and Wilcoxon Test	SVM, DT, RF, MLP
44	the KDD CUP 99 dataset	Embedded	Accuracy, No. Selected feature, Average Runtime, and F-measure	NB, DT
45	The UCI small dataset in Biological, life and other fields	Embedded	Accuracy, No. Selected feature, Sensitivity, Specificity, Average Runtime, and Other	Other

As Table 10 shows, KNN classifier has been used in most cases (67%). SVM is in the second place (31%), and 2% is allocated to the rest of the classifiers. On the other hand, 53% of the methods have used the wrapper technique and 29% the hybrid technique. The embedded and filter techniques with 16% and 2%, respectively, are placed at the next levels. Regarding the evaluation criteria, it can be seen that nearly 96% of the papers have used accuracy, and 91% have used the number of selected features, which are then followed by Runtime,

Statistical standard deviation, Wilcoxon test, and Fitness analysis with 49%, 44%, 40%, and 29%, respectively. Sensitivity was used in 20% of the papers.
The review showed that the mentioned feature selection methods have been used in different areas of the real world, such as Classification of financial

data (Key 1), Classification of agricultural data (Key 5), the Face and voice recognition and classification (Keys 21, 23, and 25), Disease and cancer diagnosis (Keys 2, 3, 10, 11, 12, 13, 14, 16, 20, 21, 27, 28, 29, 31, 40, 42, and 45), Network traffic classification (Keys 6 and 44), Text classification (Keys 9, 17, and 21), and Human activity recognition (Key 7).

5. Conclusions

Today, due to the growth of high-dimensional datasets and challenges in pattern recognition processes, machine learning techniques, data mining, and natural language processing, a lot of research has been done to select the best features among all available ones. The feature selection process has improved learning speed, rule simplicity, data visualization, and prediction accuracy. This paper reviewed a total of 45 different feature selection methods based on 9 optimization algorithms (proposed from 2018 to 2021). The findings showed that most of the optimization methods (more than 90%) have sought to maximize the classification accuracy and minimize the number of selected features by checking the fitness of the obtained solutions. Although, the objectives of feature selection are different in different applications. For example, in medical datasets, most researches may look for the minimum number of features with the highest level of accuracy; however, in other datasets such as time series datasets, where the response variable is generally continuous and time plays an important role in the response variable, the goal is not finding minimum features and reaching maximum accuracy since it may lead to overfitting. Therefore, researchers generally search for datasets with which they can generalize the entire dataset. One of the most famous time series datasets are agriculture datasets. Additionally, it was observed that more than half of these methods are wrapper-based methods, which become computationally expensive as the number of features increases. Therefore, hybrid feature selection methods have been developed in the literature, which use filter methods to remove redundant and irrelevant features and then use wrapper methods to further refine the selected subset. Nearly one-third of the investigated methods were hybrid feature selection methods. In addition, it was found that the most popular classifier used in the investigated methods is KNN with more than two-thirds of cases, followed by SVM with nearly one-third of cases.
Despite the great power of optimization algorithms in solving the feature selection problem, there is not any single most effective and perfect optimization algorithm in feature selection. Therefore, the followings are expected for future trends of meta-heuristic algorithms in the feature selection domain:

1. A comprehensive study on optimization algorithms that have shown promising feature selection results in high-dimensional datasets.
2. The development and use of parallel optimization methods for faster and more accurate feature selection.
3. The exploration of a unified measurement criterion to evaluate the performance, advantages, and disadvantages of the studied algorithms.

References

[1] A.-D. Li, B. Xue, and M. Zhang, "Improved binary particle swarm optimization for feature selection with new initialization and search space reduction strategies," Applied Soft Computing, vol. 106, p. 107302, 2021.
[2] H. B. Nguyen, B. Xue, and P. Andreae, "A hybrid GA-GP method for feature reduction in classification," in Asia-Pacific Conference on Simulated Evolution and Learning, 2017, pp. 591-604: Springer.
[3] B. Xue and G. Chen, "Guest editorial: special issue on evolutionary optimization, feature reduction and learning," ed: Springer, 2016.
[4] H. Xu, B. Xue, and M. Zhang, "A duplication analysis-based evolutionary algorithm for biobjective feature selection," IEEE Transactions on Evolutionary Computation, vol. 25, no. 2, pp. 205-218, 2020.
[5] Y. Bi, B. Xue, and M. Zhang, "Evolving deep forest with automatic feature extraction for image classification using genetic programming," in International Conference on Parallel Problem Solving from Nature, 2020, pp. 3-18: Springer.

[6] Y. Bi, B. Xue, and M. Zhang, "Evolving deep forest with automatic feature extraction for image classification using genetic programming," in International Conference on Parallel Problem Solving from Nature, 2020, pp. 3-18: Springer.

[7] B. Peng, S. Wan, Y. Bi, B. Xue, and M. Zhang, "Automatic Feature Extraction and Construction Using Genetic Programming for Rotating Machinery Fault Diagnosis," IEEE Transactions on Cybernetics, 2020.

[8] M. Rostami, K. Berahmand, E. Nasiri, and S. Forouzande, "Review of swarm intelligence-based feature selection methods," Engineering Applications of Artificial Intelligence, vol. 100, p. 104210, 2021.

[9] M. Hammami, S. Bechikh, A. Louati, M. Makhlouf, and L. B. Said, "Feature construction as a bi-level optimization problem," Neural Computing and Applications, vol. 32, no. 17, pp. 13783-13804, 2020.

[10] M. Hammami, S. Bechikh, M. Makhlouf, C.-C. Hung, and L. B. Said, "Class dependent feature construction as a bi-level optimization problem," in 2020 IEEE Congress on Evolutionary Computation (CEC), 2020, pp. 1-8: IEEE.

[11] Q. Fan, Y. Bi, B. Xue, and M. Zhang, "Genetic programming for feature extraction and construction in image classification," Applied Soft Computing, vol. 118, p. 108509, 2022.

[12] B. Peng, S. Wan, Y. Bi, B. Xue, and M. Zhang, "Automatic feature extraction and construction using genetic programming for rotating machinery fault diagnosis," IEEE transactions on cybernetics, vol. 51, no. 10, pp. 4909-4923, 2020.

[13] Q. U. Ain, B. Xue, H. Al-Sahaf, and M. Zhang, "Genetic programming for feature selection and feature construction in skin cancer image classification," in Pacific rim international conference on artificial intelligence, 2018, pp. 732-745: Springer.

[14] I. M. R. Albuquerque, B. H. Nguyen, B. Xue, and M. Zhang, "A Novel Genetic Algorithm Approach to Simultaneous Feature Selection and Instance Selection," in 2020 IEEE Symposium Series on Computational Intelligence (SSCI), 2020, pp. 616-623: IEEE.

[15] M. Hammami, S. Bechikh, C.-C. Hung, and L. B. Said, "A multi-objective hybrid filter-wrapper evolutionary approach for feature selection," Memetic Computing, vol. 11, no. 2, pp. 193-208, 2019.

[16] M. Hammami, S. Bechikh, C.-C. Hung, and L. B. Said, "Class-Dependent Weighted Feature Selection as a Bi-Level Optimization Problem," in International Conference on Neural Information Processing, 2020, pp. 269-278: Springer.

[17] G.-G. Wang, L. Guo, A. H. Gandomi, G.-S. Hao, and H. Wang, "Chaotic krill herd algorithm," Information Sciences, vol. 274, pp. 17-34, 2014.

[18] B. Xue and M. Zhang, "Evolutionary computation for feature selection and feature construction," in Proceedings of the 2020 Genetic and Evolutionary Computation Conference Companion, 2020, pp. 1283-1312

[19] Y. Xue, B. Xue, and M. Zhang, "Self-adaptive particle swarm optimization for large-scale feature selection in classification," ACM Transactions on Knowledge Discovery from Data (TKDD), vol. 13, no. 5, pp. 1-27, 2019.

[20] Y. Bi, B. Xue, and M. Zhang, "Multi-objective genetic programming for feature learning in face recognition," Applied Soft Computing, vol. 103, p. 107152, 2021.

[21] M. Hammami, S. Bechikh, C.-C. Hung, and L. B. Said, "A multi-objective hybrid filter-wrapper evolutionary approach for feature construction on high-dimensional data," in 2018 IEEE Congress on Evolutionary Computation (CEC), 2018, pp. 1-8: IEEE.

[22] V. Kothari, J. Anuradha, S. Shah, and P. Mittal, "A survey on particle swarm optimization in feature selection," in International Conference on Computing and Communication Systems, 2011, pp. 192-201: Springer.

[23] B. Xue, M. Zhang, W. N. Browne, and X. Yao, "A survey on evolutionary computation approaches to feature selection," IEEE Transactions on Evolutionary Computation, vol. 20, no. 4, pp. 606-626, 2015.

[24] L. Brezočnik, I. Fister, and V. Podgorelec, "Swarm intelligence algorithms for feature selection: a review," Applied Sciences, vol. 8, no. 9, p. 1521, 2018.

[25] B. H. Nguyen, B. Xue, and M. Zhang, "A survey on swarm intelligence approaches to feature selection in data mining," Swarm and Evolutionary Computation, vol. 54, p. 100663, 2020.

[26] M. Dash and H. Liu, "Feature selection for classification," Intelligent data analysis, vol. 1, no. 3, pp. 131-156, 1997.

[27] R. Kohavi and G. H. John, "Wrappers for feature subset selection," Artificial intelligence, vol. 97, no. 1-2, pp. 273-324, 1997.

[28] A. Abdolmaleki and M. H. Rezvani, "An optimal context-aware content-based movie recommender system using genetic algorithm: a case study on MovieLens dataset," Journal of Experimental & Theoretical Artificial Intelligence, pp. 1-27, 2022.

[29] J. H. Holland, "Genetic algorithms and the optimal allocation of trials," SIAM Journal on Computing, vol. 2, no. 2, pp. 88-105, 1973.

[30] D. E. Goldberg and J. H. Holland, "Genetic algorithms and machine learning," 1988.

[31] D. Robinson et al., "Genetic Algorithm for Feature and Latent Variable Selection for Nutrient Assessment in Horticultural Products," in 2021

IEEE Congress on Evolutionary Computation (CEC), 2021, pp. 272-279: IEEE.
[32] H. Dong, T. Li, R. Ding, and J. Sun, "A novel hybrid genetic algorithm with granular information for feature selection and optimization," Applied Soft Computing, vol. 65, pp. 33-46, 2018.
[33] X.-Y. Liu, Y. Liang, S. Wang, Z.-Y. Yang, and H.-S. Ye, "A hybrid genetic algorithm with wrapper-embedded approaches for feature selection," IEEE Access, vol. 6, pp. 22863-22874, 2018.
[34] C. B. Gokulnath and S. Shantharajah, "An optimized feature selection based on genetic approach and support vector machine for heart disease," Cluster Computing, vol. 22, no. 6, pp. 14777-14787, 2019.
[35] R. Guha et al., "Deluge based Genetic Algorithm for feature selection," Evolutionary intelligence, pp. 1-11, 2019.
[36] G. Dueck, "New optimization heuristics: The great deluge algorithm and the record-to-record travel," Journal of Computational physics, vol. 104, no. 1, pp. 86-92, 1993.
[37] F. Amini and G. Hu, "A two-layer feature selection method using genetic algorithm and elastic net," Expert Systems with Applications, vol. 166, p. 114072, 2021.
[38] M. Dorigo and L. M. Gambardella, "Ant colony system: a cooperative learning approach to the traveling salesman problem," IEEE Transactions on evolutionary computation, vol. 1, no. 1, pp. 53-66, 1997.
[39] A. Khan and A. R. Baig, "Multi-objective feature subset selection using mRMR based enhanced ant colony optimization algorithm (mRMR-EACO)," Journal of Experimental & Theoretical Artificial Intelligence, vol. 28, no. 6, pp. 1061-1073, 2016.
[40] Y. Wan, M. Wang, Z. Ye, and X. Lai, "A feature selection method based on modified binary coded ant colony optimization algorithm," Applied Soft Computing, vol. 49, pp. 248-258, 2016.
[41] S. Saremi, S. Mirjalili, and A. Lewis, "Grasshopper optimisation algorithm: theory and application," Advances in Engineering Software, vol. 105, pp. 30-47, 2017.
[42] H. Peng, C. Ying, S. Tan, B. Hu, and Z. Sun, "An improved feature selection algorithm based on ant colony optimization," IEEE Access, vol. 6, pp. 69203-69209, 2018.
[43] H. Ghimatgar, K. Kazemi, M. S. Helfroush, and A. Aarabi, "An improved feature selection algorithm based on graph clustering and ant colony optimization," Knowledge-Based Systems, vol. 159, pp. 270-285, 2018.
[44] P. Moradi and M. Rostami, "Integration of graph clustering with ant colony optimization for feature selection," Knowledge-Based Systems, vol. 84, pp. 144-161, 2015.
[45] M. Ghosh, R. Guha, R. Sarkar, and A. Abraham, "A wrapper-filter feature selection technique based on ant colony optimization," Neural Computing and Applications, pp. 1-19, 2019.
[46] R. J. Manoj, M. A. Praveena, and K. Vijayakumar, "An ACO–ANN based feature selection algorithm for big data," Cluster Computing, vol. 22, no. 2, pp. 3953-3960, 2019.
[47] M. Mafarja et al., "Evolutionary population dynamics and grasshopper optimization approaches for feature selection problems," Knowledge-Based Systems, vol. 145, pp. 25-45, 2018.
[48] A. Lewis, S. Mostaghim, and M. Randall, "Evolutionary population dynamics and multi-objective optimisation problems," in Multi-Objective optimization in computational intelligence: theory and practice: IGI Global, 2008, pp. 185-206.
[49] M. Mafarja, I. Aljarah, H. Faris, A. I. Hammouri, A.-Z. Ala'M, and S. Mirjalili, "Binary grasshopper optimisation algorithm approaches for feature selection problems," Expert Systems with Applications, vol. 117, pp. 267-286, 2019.
[50] J. Kennedy and R. C. Eberhart, "A discrete binary version of the particle swarm algorithm," in 1997 IEEE International conference on systems, man, and cybernetics. Computational cybernetics and simulation, 1997, vol. 5, pp. 4104-4108: IEEE.
[51] E. Rashedi, H. Nezamabadi-Pour, and S. Saryazdi, "BGSA: binary gravitational search algorithm," Natural Computing, vol. 9, no. 3, pp. 727-745, 2010.
[52] H. Hichem, M. Elkamel, M. Rafik, M. T. Mesaaoud, and C. Ouahiba, "A new binary grasshopper optimization algorithm for feature selection problem," Journal of King Saud University-Computer and Information Sciences, 2019.
[53] H. T. Ibrahim, W. J. Mazher, O. N. Ucan, and O. Bayat, "A grasshopper optimizer approach for feature selection and optimizing SVM parameters utilizing real biomedical data sets," Neural Computing and Applications, vol. 31, no. 10, pp. 5965-5974, 2019.
[54] A. Zakeri and A. Hokmabadi, "Efficient feature selection method using real-valued grasshopper optimization algorithm," Expert Systems with Applications, vol. 119, pp. 61-72, 2019.
[55] R. Eberhart and J. Kennedy, "Particle swarm optimization," in Proceedings of the IEEE international conference on neural networks, 1995, vol. 4, pp. 1942-1948: Citeseer.
[56] B. Tran, B. Xue, and M. Zhang, "A new representation in PSO for discretization-based feature selection," IEEE Transactions on Cybernetics, vol. 48, no. 6, pp. 1733-1746, 2017.
[57] B. Tran, M. Zhang, and B. Xue, "A PSO based hybrid feature selection algorithm for high-dimensional classification," in 2016 IEEE congress on evolutionary computation (CEC), 2016, pp. 3801-3808: IEEE.

[58] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," Advances in engineering software, vol. 69, pp. 46-61, 2014.

[59] E. Emary, H. M. Zawbaa, and A. E. Hassanien, "Binary grey wolf optimization approaches for feature selection," Neurocomputing, vol. 172, pp. 371-381, 2016.

[60] A. A. Chowdhury, V. S. Borkar, and G. K. Birajdar, "Indian language identification using time-frequency image textural descriptors and GWO-based feature selection," Journal of Experimental & Theoretical Artificial Intelligence, vol. 32, no. 1, pp. 111-132, 2020.

[61] 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.

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

[63] S. Rajamohana and K. Umamaheswari, "Hybrid approach of improved binary particle swarm optimization and shuffled frog leaping for feature selection," Computers & Electrical Engineering, vol. 67, pp. 497-508, 2018.

[64] M. Eusuff, K. Lansey, and F. Pasha, "Shuffled frog-leaping algorithm: a memetic meta-heuristic for discrete optimization," Engineering optimization, vol. 38, no. 2, pp. 129-154, 2006.

[65] A. P. Engelbrecht, J. Grobler, and J. Langeveld, "Set based particle swarm optimization for the feature selection problem," Engineering Applications of Artificial Intelligence, vol. 85, pp. 324-336, 2019.

[66] J. Langeveld and A. P. Engelbrecht, "Set-based particle swarm optimization applied to the multidimensional knapsack problem," Swarm Intelligence, vol. 6, no. 4, pp. 297-342, 2012.

[67] 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.

[68] S. Mirjalili and A. Lewis, "The whale optimization algorithm," Advances in engineering software, vol. 95, pp. 51-67, 2016.

[69] L. Kumar and K. K. Bharti, "An improved BPSO algorithm for feature selection," in Recent trends in communication, computing, and electronics: Springer, 2019, pp. 505-513.

[70] L. Kaufman and P. J. Rousseeuw, Finding groups in data: an introduction to cluster analysis. John Wiley & Sons, 2009.

[71] Y. Zhang, H.-G. Li, Q. Wang, and C. Peng, "A filter-based bare-bone particle swarm optimization algorithm for unsupervised feature selection," Applied Intelligence, vol. 49, no. 8, pp. 2889-2898, 2019.

[72] Y. Wang, J. Wang, H. Liao, and H. Chen, "Unsupervised feature selection based on Markov blanket and particle swarm optimization," Journal of Systems Engineering and Electronics, vol. 28, no. 1, pp. 151-161, 2017.

[73] K. Tadist, F. Mrabti, N. S. Nikolov, A. Zahi, and S. Najah, "SDPSO: Spark Distributed PSO-based approach for feature selection and cancer disease prognosis," Journal of Big Data, vol. 8, no. 1, pp. 1-22, 2021.

[74] Y. Pathak, K. Arya, and S. Tiwari, "Feature selection for image steganalysis using levy flight-based grey wolf optimization," Multimedia Tools and Applications, vol. 78, no. 2, pp. 1473-1494, 2019.

[75] A. Chechkin, R. Metzler, J. Klafter, and V. Y. Gonchar, "Anomalous Transport: Foundations and Applications," Introduction to the Theory of Lévy Flights, 2008.

[76] Q. Al-Tashi, S. J. A. Kadir, H. M. Rais, S. Mirjalili, and H. Alhussian, "Binary optimization using hybrid grey wolf optimization for feature selection," IEEE Access, vol. 7, pp. 39496-39508, 2019.

[77] N. Singh and S. Singh, "Hybrid algorithm of particle swarm optimization and grey wolf optimizer for improving convergence performance," Journal of Applied Mathematics, vol. 2017, 2017.

[78] P. Sharma, S. Sundaram, M. Sharma, A. Sharma, and D. Gupta, "Diagnosis of Parkinson's disease using modified grey wolf optimization," Cognitive Systems Research, vol. 54, pp. 100-115, 2019.

[79] Q. Tu, X. Chen, and X. Liu, "Multi-strategy ensemble grey wolf optimizer and its application to feature selection," Applied Soft Computing, vol. 76, pp. 16-30, 2019.

[80] S. Arora, H. Singh, M. Sharma, S. Sharma, and P. Anand, "A new hybrid algorithm based on grey wolf optimization and crow search algorithm for unconstrained function optimization and feature selection," Ieee Access, vol. 7, pp. 26343-26361, 2019.

[81] M. Mafarja, A. Qasem, A. A. Heidari, I. Aljarah, H. Faris, and S. Mirjalili, "Efficient hybrid nature-inspired binary optimizers for feature selection," Cognitive Computation, vol. 12, no. 1, pp. 150-175, 2020.

[82] 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, p. 112824, 2020.

[83] S. Arora and S. Singh, "Node localization in wireless sensor networks using butterfly optimization algorithm," Arabian Journal for Science and Engineering, vol. 42, no. 8, pp. 3325-3335, 2017.

[84] S. Mirjalili, A. H. Gandomi, S. Z. Mirjalili, S. Saremi, H. Faris, and S. M. Mirjalili, "Salp Swarm

Algorithm: A bio-inspired optimizer for engineering design problems," Advances in Engineering Software, vol. 114, pp. 163-191, 2017.

[85] S. Arora and P. Anand, "Binary butterfly optimization approaches for feature selection," Expert Systems with Applications, vol. 116, pp. 147-160, 2019.

[86] Z. Sadeghian, E. Akbari, and H. Nematzadeh, "A hybrid feature selection method based on information theory and binary butterfly optimization algorithm," Engineering Applications of Artificial Intelligence, vol. 97, p. 104079, 2021.

[87] W. Gao, L. Hu, P. Zhang, and F. Wang, "Feature selection by integrating two groups of feature evaluation criteria," Expert Systems with Applications, vol. 110, pp. 11-19, 2018.

[88] H. Faris et al., "An efficient binary salp swarm algorithm with crossover scheme for feature selection problems," Knowledge-Based Systems, vol. 154, pp. 43-67, 2018.

[89] I. Aljarah, M. Mafarja, A. A. Heidari, H. Faris, Y. Zhang, and S. Mirjalili, "Asynchronous accelerating multi-leader salp chains for feature selection," Applied Soft Computing, vol. 71, pp. 964-979, 2018.

[90] A. E. Hegazy, M. Makhlouf, and G. S. El-Tawel, "Feature selection using chaotic salp swarm algorithm for data classification," Arabian Journal for Science and Engineering, vol. 44, no. 4, pp. 3801-3816, 2019.

[91] R. A. Ibrahim, A. A. Ewees, D. Oliva, M. Abd Elaziz, and S. Lu, "Improved salp swarm algorithm based on particle swarm optimization for feature selection," Journal of Ambient Intelligence and Humanized Computing, vol. 10, no. 8, pp. 3155-3169, 2019.

[92] A. E. Hegazy, M. Makhlouf, and G. S. El-Tawel, "Improved salp swarm algorithm for feature selection," Journal of King Saud University-Computer and Information Sciences, vol. 32, no. 3, pp. 335-344, 2020.

[93] M. Tubishat et al., "Dynamic salp swarm algorithm for feature selection," Expert Systems with Applications, vol. 164, p. 113873, 2021.

[94] Y. Zheng et al., "A novel hybrid algorithm for feature selection based on whale optimization algorithm," IEEE Access, vol. 7, pp. 14908-14923, 2018.

[95] M. Tubishat, M. A. Abushariah, N. Idris, and I. Aljarah, "Improved whale optimization algorithm for feature selection in Arabic sentiment analysis," Applied Intelligence, vol. 49, no. 5, pp. 1688-1707, 2019.

[96] R. Storn and K. Price, "Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces," Journal of global optimization, vol. 11, no. 4, pp. 341-359, 1997.

[97] H. Nematzadeh, R. Enayatifar, M. Mahmud, and E. Akbari, "Frequency based feature selection method using whale algorithm," Genomics, vol. 111, no. 6, pp. 1946-1955, 2019.

[98] A. G. Hussien, A. E. Hassanien, E. H. Houssein, S. Bhattacharyya, and M. Amin, "S-shaped binary whale optimization algorithm for feature selection," in Recent trends in signal and image processing: Springer, 2019, pp. 79-87.

[99] X.-S. Yang, "Firefly algorithm, Levy flights and global optimization," in Research and development in intelligent systems XXVI: Springer, 2010, pp. 209-218.

[100] R. Sawhney, P. Mathur, and R. Shankar, "A firefly algorithm based wrapper-penalty feature selection method for cancer diagnosis," in International Conference on Computational Science and Its Applications, 2018, pp. 438-449: Springer.

[101] L. Zhang, K. Mistry, C. P. Lim, and S. C. Neoh, "Feature selection using firefly optimization for classification and regression models," Decision Support Systems, vol. 106, pp. 64-85, 2018.