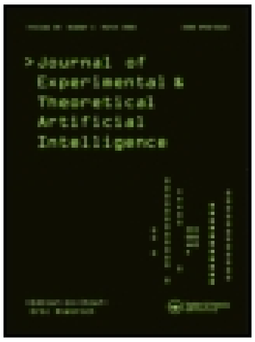

Journal of Experimental & Theoretical Artificial Intelligence

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/teta20

A review of feature selection methods based on meta-heuristic algorithms

Zohre Sadeghian, Ebrahim Akbari, Hossein Nematzadeh & Homayun Motameni

To cite this article: Zohre Sadeghian, Ebrahim Akbari, Hossein Nematzadeh & Homayun Motameni (2023): A review of feature selection methods based on meta-heuristic algorithms, Journal of Experimental & Theoretical Artificial Intelligence, DOI: 10.1080/0952813X.2023.2183267

To link to this article: https://doi.org/10.1080/0952813X.2023.2183267

Published online: 27 Feb 2023.

JOURNAL OF EXPERIMENTAL & THEORETICAL ARTIFICIAL INTELLIGENCE
https://doi.org/10.1080/0952813X.2023.2183267

Taylor & Francis
Taylor & Francis Group

ARTICLE

A review of feature selection methods based on meta-heuristic algorithms

Zohre Sadeghian, Ebrahim Akbari, Hossein Nematzadeh and Homayun Motameni

Department of Computer Engineering, Sari Branch, Islamic Azad University, Sari, Iran

ABSTRACT
Feature selection is a real-world problem that finds a minimal feature subset from an original feature set. A good feature selection method, in addition to selecting the most relevant features with less redundancy, can also reduce computational costs and increase classification performance. One of the feature selection approaches is using meta-heuristic algorithms. This work provides a summary of some meta-heuristic feature selection methods proposed from 2018 to 2022 that were designed and implemented on a wide range of different data for solving feature selection problem. Evaluation criteria, fitness functions and classifiers used and the time complexity of each method are also depicted. 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 dataset, a special meta-heuristic algorithm should be used. The results of this study and the identified research gaps can be used by researchers in this field.

ARTICLE HISTORY
Received 16 June 2021
Accepted 16 February 2023

KEYWORDS
Data dimension reduction; classification; feature selection; optimisation algorithm; meta-heuristic algorithms

Introduction

Lot of information has been obtained in many sciences with the advancement of data collection and storage in recent decades. Researchers in different fields such as engineering, astronomy, biology, economics, etc. are facing more and more observations every day. Today's the increasing size of data has become a major issue in data mining. In the past, classical datasets with a maximum of a few dozen features have had acceptable classification accuracy (Mullick et al., 2020). Today, we are faced with big data and high-dimensional data in which the number of features has increased dramatically (Al-Helali et al., 2019; Pei et al., 2018; Tran et al., 2019). This increase in dimensions increases the computational cost of the system, hence leading to a reduction in the classification accuracy rate. In addition, as the number of features grows, the need for more data samples surges, which, in turn, increases the temporal and spatial complexity of the problem. Therefore, it can be concluded that traditional statistical methods have lost their effectiveness today for two reasons: The increase in the number of observations, and the increase in the number of features related to an observation. In fact, in most cases, all features of data are not important to find the knowledge that lies in the data. For this reason, in many areas of study, reducing the size of the data is one of the significant issues. Data dimension reduction methods are generally divided into two categories (A. -D. Li et al., 2021; B. Xue & Chen, 2016; H. B. Nguyen et al., 2017; Xu et al., 2020):

CONTACT Ebrahim Akbari ebrahimakbari30@yahoo.com Department of Computer Engineering, Sari Branch, Islamic Azad University, Sari, Iran

- Feature Extraction: These methods map a multidimensional space to a smaller space. In fact, by combining the values of existing features, they create fewer features so that these features contain all (or most) of the information contained in the original features (Bi et al., 2018, Bi et al., 2020a, B. Peng et al., 2020a; Hammami et al., 2020; Hammami et al., 2020c, Fan et al., 2022; B. Peng et al., 2020b; Rostami et al., 2021).
- Feature Selection: These methods try to reduce the size of the data by selecting a subset of the primary features (Ain et al., 2018; Albuquerque et al., 2020, Hammami et al., 2019; Hammami et al., 2020a, G. -G. Wang et al., 2014; B. Xue & Zhang, 2020; Y. Xue et al., 2019; Kashani & Hamidzadeh, 2020).

In fact, feature extraction creates a set of new features. Therefore, we cannot have a proper physical understanding of these features. In contrast, feature selection preserves the physical perception of the original features with retaining some of the main features, and presents better readability and interpretability of the models. Therefore, feature selection is often preferred in many areas (Bi et al., 2021; Hammami et al., 2020a). In recent years, several reviews have been presented to study different feature selection methods for classification and clustering purposes (Deng et al., 2019; Hancer et al., 2020; Remeseiro & Bolon-Canedo, 2019). In this regard, optimisation-based feature selection methods have been studied extensively in recent years and have been successful to a great extent (Hammami et al., 2018, 2019). Numerous review papers have been presented to study and review this type of methods. For example, (Kothari et al., 2011) studied 17 the Particle Swarm Optimisation (PSO)-based feature selection methods that were published before 2010. (B. Xue et al., 2015) studied Evolutionary Computation (EC)-based feature selection methods. However, this study was limited to Genetic Algorithm (GA), Genetic programming (GP), PSO and Ant Colony Optimisation (ACO)-based methods published before 2015. (Brezočnik et al., 2018) studied the Swarm Intelligence (SI)-based feature selection methods proposed between 2001 and 2017. They classified these methods based on the initialisation and search mechanism. Then, (B. H. Nguyen et al., 2020) categorised SI-based methods based on the representation and the search mechanism. This work limited to PSO, Artificial Bee Colony (ABC) and ACO-based feature selection methods before 2018. (M. Sharma & Kaur, 2021) presented an analysis of meta-heuristic methods inspired by nature in 2018 and 2019. In this work, the criteria of redundancy and relevance, scalability and the use of binary and chaotic types of these methods were mentioned. However, this study did not consider purely filter-based and wrapper-based feature selection methods. Also, the fitness function used by these methods was not discussed. As it is known, the above articles are limited to the years before 2019 and some specific algorithms or areas. And in some works, criteria such as fitness function and used classifiers have not been investigated. Accordingly, this article reviews and compares feature selection methods based on 21 different meta-heuristic algorithms in different areas between 2018 and 2022.

The major contributions of this article can be summarised as:

- Introducing the new feature selection methods based on meta-heuristic algorithms from 2018 to 2022.
- Review of performance evaluation criteria of methods
- Review and comparison of the fitness function used in these methods
- Review of frequency of meta-heuristic algorithms used in the studied methods
- Review and comparison of time complexity of methods

The remainder of this paper is organised as follows: Section 2 describes the basic concept of feature selection and discusses a variety of different methods proposed in the literature for solving this problem. Section 3 reviews the related work based on the meta-heuristic algorithms proposed between 2018 and 2022. Section 4 provides an analysis of these methods. Finally, the research conclusion is presented in Section 5.

Feature selection

As a simple definition, the feature selection refers to finding a subset of features with the minimum size, containing the necessary and sufficient information for the intended purpose. Different feature selection methods have attempted to find the best subset from among 2^N candidate subsets. But, due to the breadth of possible results, finding the optimal subset is difficult and very expensive in medium and large values of N.

The feature selection process consists of four following basic steps (Dash & H, 1997).

- Subset generation: A feature subset is generated by a search method.
- Subset evaluation: Each produced subset is required to be evaluated based on a special criterion to determine how desirable it is.
- Stopping criterion: The feature selection process must be stopped under a certain criterion such as achieving a certain number of features, reaching the maximum number of iterations, producing an optimal subset, or not producing better subsets by adding or removing a new feature.
- Result validation: The feature subset is tested to check whether it has the necessary conditions to solve the desired problem or not?

Figure 1 shows the steps of the feature selection process.

Feature selection methods can be examined and classified from different perspectives. Depending on feature selection mechanism, feature selection methods can be broadly classified into three categories: filter, wrapper, and embedded (Kohavi & John, 1997). Filter-based methods calculate a rank or a score for each feature based on data dependency techniques. Then features with lower scores are omitted. The advantage of these models is the low computational cost, good generalisation ability, high speed and applicable to high-dimensional data. However, they may not be suitable for target learning algorithms due to the lack of a learning algorithm in search phase. On the other hand, the wrapper methods include a learning algorithm as a black box and use its predictive performance to assess the selected features. These methods consist of 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 due to the use of machine learning algorithms. The embedded methods perform feature selection in the training process and are usually applied to specific learning machines (Sadeghian et al., 2021). In the first stage of these methods, the filter-based methods are applied to reduce data size, and then in the second stage, the wrapper methods are applied to select the best feature subset. Thus, the embedded methods inherit the advantages of both filter and wrapper methods. They have better performance than the wrapper methods because they do not require re-evaluation of feature sets. The embedded feature selection methods eliminate redundant and irrelevant

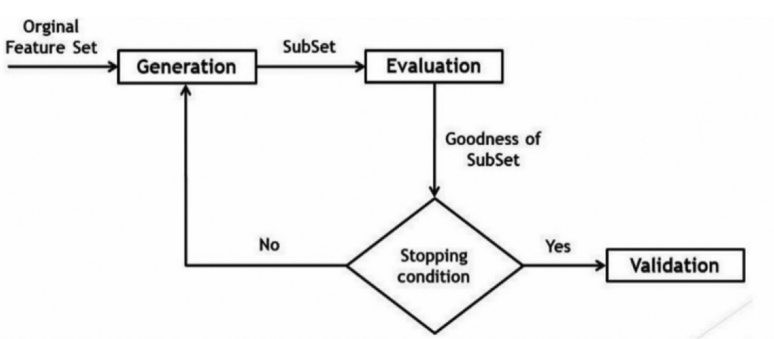

Figure 1. Feature selection process (Dash & H, 1997).

features without considerably decreasing the speed or increasing the computational complexity. The majority of meta-heuristic algorithms are considered as wrapper method because they generate a set of solutions during a given iteration and then evaluate them at each iteration, and finally, they extract the best solution. Based on the techniques adopted, the feature selection methods are categorised into the following five groups (J. Li et al., 2017):

- Similarity-based methods: These methods work on the basis of data similarity that can be extracted from class labels in supervised methods or can be calculated using different distance metric measurements in unsupervised methods.
- Information theory-based methods: These methods consider feature relevancy and feature redundancy in selection process.
- Sparse learning-based methods: The purpose of these methods is minimising fitting errors along with sparse adjustment terms.
- Statistical-based methods: These methods use different statistical criteria, such as the concept of variance, T-score, chi-square score, Gini index, etc. to select relevant features or remove irrelevant features.
- Other methods: Feature selection methods that do not belong to the four first categories, e.g. hybrid feature selection methods, deep learning based methods, and meta-heuristic feature selection algorithms are classified in last category. Meta-heuristic search algorithms have been inspired by biological processes in nature. These algorithms can be used for solving feature selection problem, with a little modification. Recent research has shown that these methods are more effective for large data sets (Hancer et al., 2022; Pei et al., 2022; Ain et al., 2022; Pei et al., 2021; A. -D. Li et al., 2021; Bi et al., 2020, Bi et al., 2020a, Hamidzadeh, 2021).

Meta-heuristic algorithms for feature selection

Nowadays, with the rapid growth of real-world problems and the importance of access speed to answers, classical methods cannot deal with many problems and random algorithms are mostly used. Therefore, in recent decades, the use of meta-heuristic algorithms has grown significantly. Unlike classical methods, meta-heuristic 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. Meta-heuristic algorithms are divided into population-based and path-based methods based on their search strategies. The Genetic Algorithm that uses a set of chromosomes or whale and cuckoo algorithms that use several agents are population-based algorithms. These methods consider ways to coalesce and extend the elements of the solutions that already exist. On the other hand, path-based algorithms, e.g., Tabu Search (TS) and Simulated Annealing (SA), incorporate strategies to transform a single solution (Rupasinghe et al., 2010). In this section, the feature selection studies which have used meta-heuristic algorithms and have published between 2018 and 2022, are reviewed.

Genetic Algorithm-based feature selection

Genetic Algorithm (GA) is a special type of evolutionary algorithms, that uses the technique of inheritance and mutation. First, it was introduced by (Holland, 1973) and its position was consolidated by the efforts of (Goldberg & Holland, 1988). GA uses the mechanism of the superior solution (chromosome) selection in each generation. This algorithm is based on two principles: the survival of the best and mating. In recent years, various methods based on this algorithm have been proposed to solve the feature selection problem, such as: GIFS (Abdel-Basset et al., 2020), GA-enhanced PLSR (Robinson et al., 2021). Table 1 demonstrates the primary details taken from the chosen recently-published GA-based papers.

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

Publication	Methodology Fitness function Stopping Criteria Experimental results	Findings	Key NO.
(Dong et al., 2018)	This method employs 3 strategies: Improved Binary GA with Feature Granulation (IBGAFG) (combination of granular computing theory with GA) 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 Optimisation-based GA (ROGA) to obtain the optimal granularity parameters. Each chromosome in IBGAFG includes two parts. The first part is a binary vector that correspond to a feature subset. The value 1 and 0 are used for representing the selected feature and the not selected feature, respectively. The second Part consists of one bit for representing the granulation of the current chromosome. Classification accuracy The maximum number of iterations (500) Superiority of this method over the Fast Correlation-Based Filter (FCBF), Best Incremental Ranked Subset (BIRS), GA, Markov Blanket-Embedded GA (MBEGA), Maximum Relevancy- Minimum Redundancy (MRMR), and INRSG.	1) Obtaining granular parameters by self-adaptive manner, 2) Proving the effectiveness of IBGAFG for large-scale data.	1
(X. -Y. Liu et al., 2018)	The Hybrid Genetic Algorithm with Wrapper-Embedded feature selection (HGAWE) was proposed by combining genetic operations (global search) and Hybrid $L_{1/2} + L_2$ Regularisation (HLR) embedded method (local search). In HGAWE, a new chromosome representation is used that includes intron (the penalised control parameters) and exon (the coefficients of the features in the learning model). Classification accuracy Reaching the number of specific features (2000) Superiority of this method over the GA, Genetic Programming (GP), Memetic Algorithm (MA), Smoothly Clipped Absolute Deviation L2 (SCAD-L2), Elastic net, and HLR on some simulation data and five gene microarray data sets.	1) Selecting effectively the relevant features, predicting the patients' class, and constructing the accurate learning model in high-dimensional biological datasets. 2) A practical tool for learning prediction.	2

(*Continued*)

Table 1. (Continued).

Publication	Methodology Fitness function Stopping Criteria Experimental results	Findings	Key NO.
(Gokulnath and Shantharajah, 2019	GA-SVM uses an optimisation function based on Support Vector Machine (SVM) in GA to select more significant features. GA uses a single point crossover for mating of two chromosomes, and a mutation operator to change a bit from 0 to 1 and 1 to 0. The mutation and crossover are used for searching the entire search space and finding better solution from the current solution, respectively. Each chromosome has p genes, where *p* is the number of original features in dataset. Each gene can accept one of the values of 0 (non-selected feature) or 1 (selected feature). SVM is used to calculate the fitness values of chromosomes. Classification accuracy The maximum number of iterations (10* Number of features) Superiority of this method over the Relief, CFS, Information Gain (IG), Consistency subset, Chi Squared, Gain Ratio, and GA on the dataset collected from the Cleveland heart disease database.	Achieving good performance with SVM classifier compared to Multilayer Perceptron (MLP), K Nearest Neighbour (KNN), and J48 classifiers.	3
(Guha et al., 2019)	The Deluge-based Genetic Algorithm (DGA) was proposed, which uses the Great Deluge Algorithm (GDA) (Dueck, 1993) instead of mutation operations in GA. DGA obtains a high degree of exploitation by using perturbation of candidate solutions. DGA involves five steps: Population creation, Parent selection, Uniform crossover, GDA on the child chromosomes to achieve local search, and Child replacement. In this method, each chromosome is defined similar to Key 3. DGA uses the Roulette Wheel selection (Lipowski & Lipowska, 2012) method for the selection of parent chromosomes. Classification accuracy The maximum number of iterations (20) Superiority of this method over the basic GA, Particle Swarm Optimisation (PSO), SA, and Histogram oriented Multi-Objective GA (HMOGA) on 15 standard UCI repository datasets with three classifiers, KNN, MLP (Basu et al., 2012) and SVM.	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.	4

(*Continued*)

Table 1. (Continued).

Publication	Methodology Fitness function Stopping Criteria Experimental results	Findings	Key NO.
(Amini & Hu, 2021)	A wrapper-embedded two-layer feature selection based on GA and Elastic Net (EN), called GA-EN, was proposed. In the first layer, GA is used to search the best subset of predictors with minimum prediction error and minimum computational cost. GA uses the single point crossover technique (S. Liu et al., 2013) and the mutation operator suggested by (Brown & Sumichrast, 2005). EN (Fukushima et al., 2019) is employed as a regressor in the second layer to eliminate the remaining redundant and irrelevant features.	1) Because of avoiding the exhaustive search by GA, the computation time of finding the best subset is reduced. 2) The use of EN reduces the probability of redundant/irrelevant predictors and as a result, constructs a good prediction model.	5
	$ar_{RMSE} + (1-a)w_p\|S\|/\|P\|$, where r_{RMSE} is defined based on the average root mean square error and the average of response variable. $\|S\|$ and $\|P\|$ are the number of selected predictor and the number of total predictors, respectively.		
	The maximum number of generations or the lack of fitness improvement in two consecutive generations.		
	Superiority of this method over the EN and the GA-LR on a Maize genetic dataset.		
(Abasabadi et al., 2022)	$Ga_{rank\&rand}$ is a hybrid feature selection method, that has 2 phases: 1. Sorted label Interference (SLI-γ) method (Abasabadi et al., 2021) to remove 99% of irrelevant features and 2. the wrapper GA-based feature selection to optimise the first phase solutions.	The use of SLI-γ to select 1% of the best features has a significant impact on the execution time and performance of GA in terms of classification accuracy and the number of selected features.	6
	$E(1+\beta N_s/N_t)$, where N_s and N_t are the number of selected features and the original number of features, E is classifier error and β is the cost of adding features that set to 0.52.		
	The maximum number of iterations (100)		
	Superiority of SLI-γ over ReliefF, SVM, and IG classifiers for producing better solutions and Ga over GA, GWO, and WOA for finding the optimal solution in fewer iterations on 7 high-dimensional datasets and 4 standard datasets.		

As a conclusion of the advantages and disadvantages of the studied methods, it can be said that the IBGAFG method is mostly used in the field of pattern recognition and bioinformatics. HGAWE identifies relevant genes more accurately and efficiently than other compared models, and GA-EN has a high computational cost due to further evaluations for some large problems. $Ga_{rank\&rand}$ has a low execution time due to removing 90% of irrelevant features in the first phase. But the performance of this method on multi-class datasets and non-medical datasets is unclear.

Artificial Bee Colony-based feature selection

Artificial Bee Colony (ABC) was introduced at first to optimise mathematical functions. In this algorithm, each solution represents a potential food area and agents (artificial bees) search and exploit food sources in the search space (Karaboga, 2005). This algorithm uses three types of agents: employee bees (EBs), onlooker bees (OBs), and scouts. (Hancer et al., 2015) is one of the ABC-based methods introduced in recent years. Table 2 demonstrates the primary details taken from the chosen recently-published ABC-based papers.

In conclusion, it can be said that the ABCoDT method has a higher execution time than other compared methods in some datasets.

Ant Colony Optimization-based feature selection

Ant Colony Optimisation (ACO) presented by (Dorigo & Gambardella, 1997) was inspired by the feeding behaviour of ants. Despite being blind and low intelligence, one of the ants' abilities is finding the shortest path. Ants communicate with each other based on a chemical called pheromones that they leave behind. In recent years, various methods based on this algorithm have been proposed to solve the feature selection problem, such as: MRMR Enhanced ACO (MRMR-EACO) (Khan & Baig, 2016), Modified Binary coded ACO (MBACO) (Wan et al., 2016). Table 3 demonstrates the primary details of the chosen recent ACO-based papers.

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

Publication	Methodology Fitness function Stopping Criteria Experimental results	Findings	Key NO.
(Rao et al., 2019)	Bee Colony and Gradient Boosting Decision Tree (GBDT), called ABCoDT, was proposed that has two steps: using ABC to find useful features, and using GBDT to achieve better feature subset. In fact, due to its simple structure, fewer control parameters, easy detection, and strong ability to explore the search space, ABC has been used to select the primary feature and raw data processing. Each solution in ABCoDT are defined similar to Key 3. Additionally, ABCoDT uses the regression and classification tree as a basic classifier and the coefficient GINI (Suthaharan, 2016) to determine whether branching. $SQRT\left(\sum_{i=1}^{D}(x_i - x_{centroid})^2\right)$ The maximum number of iterations Superiority of the method over the GBDT, Variance Threshold (VT), Select K Best (SKB), RFE, L12, and PCA on two breast cancer datasets and six UCI datasets.	ABCoDT provides a framework to integrate a classification method with a feature selection process.	7

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

Publication	Methodology / Fitness function / Stopping Criteria / Experimental results	Findings	Key NO.
(H. Peng et al., 2018)	The improved ACO-based Feature Selection, called FACO, combines the ACO and feature selection method. FACO uses the two-step pheromone updating rule to add pheromones to more paths which prevents ACO from falling into a local optimum. At the first stage of updating, all paths of ant colony are updated for pheromone concentrations. In second step, the pheromone concentration is enhanced in some particular paths for increasing diversity and keeping the FACO from falling into a local optimum prematurely. $aFPR + (1 - a)N_s/N_t$, where N_s is the number of selected features, and N_t is the original number of features. FPR represents the false positive rate of classification and is calculated by $FP/(TN + FP)$. In this method, the weight a was set to 0.7. The maximum number of iterations (40) Testing on the KDD CUP99 dataset and the achievement of improving the accuracy of classification, which is of great practical importance.	1) The two-stage pheromone updating method increases the search speed. 2) The path selection is more random. 3) FACO searches the feature space better.	8
(Ghimatgar et al., 2018)	TheModified Graph Clustering-based ACO (MGCACO) uses a graph to analyse the dependence between features, the Fisher Score(F-Score) to analyse the relevance of features, the absolute of Pearson's correlation to analyse the redundancy and the multiple discriminant analysis(MDA) (Moradi & Rostami, 2015) to update the amount of pheromone in steps 1 to 4. In step 5, the best features are selected by sorting descending based on the pheromone values. To give each feature a chance to be selected, MGCACO uses the multiplication instead of the subtraction performed in the GCACO method. Furthermore, MGCACO defines the pheromone value for each feature by calculating mutual information. The maximum number of iterations (50) MDA(selected feature), where MDA is multiple discriminant analysis (Moradi & Rostami, 2015). Superiority of this method over the GCACO, Laplacian Score, F-score, MRMR, ReliefF, Relevance-Redundancy Feature Selection (RRFS), and Unsupervised Feature Selection ACO (UFSACO) on seven standard UCI datasets and sleep EEG data from the Dreams Subjects Database.	Using multiplication instead of the subtraction and initialisation of pheromones in MGCACO, give a better chance to select more relevant features and thus improve the efficiency of the method.	9

(*Continued*)

Table 3. (Continued).

Publication	Methodology Fitness function Stopping Criteria Experimental results	Findings	Key NO.
(Ghosh et al., 2019)	Wrapper-Filter ACO Feature Selection (WFACOFS) was proposed, which uses a similarity-based filter method for the evaluation of selected features subsets. WFACOFS combines both Text Feature Selection ACO (TFSACO) (Wrapper method) and UFSACO (Filter method) to use their advantages. The feature set is created using the filter method and then their quality is evaluated using the wrapper method to update the pheromone values. In WFACOFS, the addition probability of a candidate feature to selected features subset is calculated based on heuristic desirability and pheromone density. The heuristic desirability is calculated based on the similarity value between last added feature and a candidate feature. Pheromone density is initialised to zero and is updated based on Local and global update values. Additionally, this method uses a fitness-based memory to maintain the best selected feature subset. $aAccuracy(S) + \beta(n - \|S\|)/n$, where $(n - \|S\|)$ is the number of unselected features. The weight of $a and \beta$ were set to 100 and 1, respectively. The maximum number of iterations (20) Superiority of this method over the SA, HMOGA, Bee Colony Optimisation Feature Selection (BCOFS), TFSACO, and Mutation-Enhanced Particle Swarm Optimisation (MEPSO) on ten UCI datasets and NIPS2003 FS challenge.	1) Restricting pheromone values to the range of [0–1] avoids premature convergence, and explores search space better. 2) By using fitness based memory, the most suited ants among all generations are get. Therefore, the effectiveness and high accuracy of the WFACOFS method are proven.	10
(Manoj et al., 2019)	ACO-ANN is a hybrid feature selection method for text categorisation. ACO, due to its ability to effectively search in the problem space, quickly congregates and effectively finds the feature subset. Accordingly, this method uses ACO to evaluate the selection process and ANN to find the best subset from the given subsets. The ANN algorithm used in ACO-ANN is the predesigned Lavenberg – Marquard Back Propagation-based ANN. The state transition rule is defined based on reducing the classification error in the output result. It should be note that the number of ants are defined based on the number of features. The ANN classification accuracy the best subset with the least classification error Superiority of this method over the IG, GA, CHI Square selection, and ACO on the Reuter's datasets.	The ACO-ANN method solves the feature selection problem efficiently on big data.	11

In conclusion, the FACO and MGCACO methods are more suitable for small and medium datasets and have a relatively better execution time than other compared methods. The WFACOFS does not consider redundancy due to the use of similarity-based filter method. For this reason, the efficiency of the method can be improved by using other filter methods instead of the similarity method. The ACO – ANN feature selection algorithm is used for textual data sets. This method should be expanded or combined with the other feature selection methods to support other types of data such as video, audio, images.

Ant Lion Optimization-based feature selection

Ant Lion Optimisation (ALO) is a stochastic optimisation algorithm proposed by (Mirjalili, 2015) and mimics the hunting behaviour of antlions in nature. ALO has no parameters to adjust and avoids local optimal stagnation due to the use of the random walking and roulette wheel. Two binary versions of ALO, called BALO-S and BALO-V were proposed by (Zawbaa et al., 2016) to solve the feature selection problem. Table 4 demonstrates the primary details of the chosen recently-published ALO-based papers.

It can be concluded that although MALO-WSVM has good and stable performance, but its performance decreases with increasing data dimensions. Also, the search capability of HBALO and LALO depend on the quality of the initial population. Therefore, the use of initialisers such as MRMR to produce the initial population is effective in improving the quality of HBALO and LALO.

Grasshopper Optimization Algorithm-based feature selection

Grasshopper Optimisation Algorithm (GOA) is an optimisation algorithm that mimics the swarming behaviour of grasshoppers that was first proposed by (Saremi et al., 2017). GOA, with the aim of improving convergence speed and avoiding local optima when searching for food, has two phases: exploration and exploitation. In the exploration phase, search agents make random moves, while in the exploitation phase, they tend to move locally and around their locations. GOA increases the average of fitness, which indicates the algorithm's ability to improve the initial random population of grasshoppers. The target fitness is also improved during iterations, which indicates the enhancement of the global optimal estimate accuracy proportional to the number of iterations. Table 5 demonstrates the primary details taken from the recent GOA-based feature selection methods.

Despite the good performance of GOA-EPD, the average CPU time of GOA-EPD is higher than the other compared meta-heuristics. The NBGOA method has achieved good results, especially for large data sets. But it needs to be expanded to support other types of datasets such as biomedicine and games. The proposed binary GOA methods do not consider redundancy despite good performance on data types which this problem can be solved by combining these methods with information theory-based methods.

Particle Swarm Optimization-based feature selection

Particle Swarm Optimisation (PSO) was first proposed by (Eberhart & Kennedy, 1995) as an uncertain optimisation method. This algorithm is inspired by the swarm movement of birds in the search of food. Each solution, called a particle, has a velocity vector to direct the movement of the particle, a position vector to identify the place of the particle and a fitness value to determine how good the particle is. In recent years, a several PSO-based feature selection methods have been proposed (Tran et al., 2016, 2017). Table 6 demonstrates the primary details taken from the recent studies conducted on the PSO-based feature selection.

The CSO reduces the search time in high-dimensional datasets. Also, the final optimal results of this method is less sensitive to initialisation. The iBPSO-SFLA is used for textual datasets. This method should be developed to support the other kind of datasets. The SBPSO and modified-BPSO are more

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

Publication	Methodology	Fitness function	Stopping Criteria	Experimental results	Findings	Key NO.
(M. Wang et al., 2019)	A feature selection method for reducing the dimensions of hyperspectral images (HSI) that is a combination of Modified ALO (MALO) and Wavelet SVM (WSVM). Due to the strong exploration ability of LF distribution, MALO uses this random walk strategy for searching the optimal solution and solving the problem of getting stuck in local optimum. The proposed method has two phase: The removing parts of the high correlation bands by using WSVM and finding the optimal band combination by using MALO.	$a \cdot Accuracy(i) + (1 - a) \cdot \log_{10} N_t / N_s$, where N_t and N_s are the original and the selected number of bands, respectively. $a = 0.9$.	Reaching the number of function evaluations (300)	Superiority of this method over the Differential Evolution (DE), Gravitational Search Algorithm (GSA), ALO, Gray Wolf Optimizer (GWO) and Cuckoo Search (CS) on the NASA EO-1 satellite dataset over the Okavango Delta, a dataset from the Kennedy Space Center (KSC) and a dataset from the AVIRIS sensor.	1) Due to being robust and adaptive and also maintaining a good balance between performance and classification accuracy, this method is for practical applications. 2) It can quickly find the best subset for some real-time applications.	12
(M. M. Mafarja & Mirjalili, 2019)	The Hybrid Binary ALO (HBALO) method was proposed by combining two incremental hill-climbing techniques (QuickReduct and CEBARKCC) and the binary ALO. QuickReduct (Jensen & Shen, 2003) and CEBARKCC (Yu et al., 2002) are two filter-based methods based on rough set and conditional entropy, respectively. Both of these algorithms simulate the forward generation method.	$d(S) * (N_t - N_s)/N_t$, where $d(S)$ is the dependency degree and the conditional entropy in the QuickReduct and CEBARKCC, respectively. N_s and N_t are the number of selected features and the original features.	The maximum number of iterations (70)	Superiority of this method over the PSO, GA, and continuous ALO on 18 UCI repository datasets.	The use of QuickReduct and CEBARKCC in the first phase of HBALO helps BALO to search the space of informative features adaptively and achieve a better balance between exploration and exploitation.	13
(Emary & Zawbaa, 2019)	Lèvy ALO (LALO) uses LF random walk (Bhandari et al., 2014) for local search. LF random walk provides several solutions separately from existing ones and solves the local minima problem and explores large search areas very efficient.	$a(1 - Accuracy) + (1 - a)\|S\|/\|N\|$, where $\|S\|$ and $\|N\|$ are the number of selected features and the original number of features, respectively. $a = 0.99$.	The maximum number of iterations (70)	Superiority of LALO over the GA, PSO, and ALO on 21 UCI datasets.	1) LALO presents repeatable results and efficient performance even with Complex fitness function. 2) The initial positions of search agents affect the performance.	14

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

Publication	Methodology Fitness function Stopping Criteria Experimental results	Findings	Key NO.
(M. Mafarja et al., 2018)	The Grasshopper Optimisation Algorithm Evolutionary Population Dynamic (GOA-EPD) was presented by combination of GOA, selection operators and Evolutionary population Dynamics (EPD) (Lewis et al., 2008) in four different form. In GOA-EPD-M and GOA-EPD-CM, each solution in the worst half updates its position based on one of the top three agents and a randomly-generated one. The GOA-EPD-M uses a simple mutation operator and the GOA-EPD-CM uses a crossover and a mutation operator. GOA-EPD-Tour and GOA-EPD-RWS approaches use the Tournament Selection (TS) (Miller & Goldberg, 1995) and Roulette Wheel Selection (RWS) (Back, 1996) operators, respectively to get a better selection in the first half of the population. Thus, solution with the lower fitness value also has an opportunity to improve the population. Like Key 13 The maximum number of iterations (100) Superiority of this method over the GA, PSO, and Binary GWO (BGWO) on 22 UCI datasets.	1) EPD has a great effect on GOA performance. 2) The selection operators improve convergence and increase the ability of the proposed method to find the best solution. 3) GOA- EPD-Tour and GOA- EPD have better performance than the other two methods.	15
(M. Mafarja et al., 2019)	Three binary versions of GOA, called BGOA-S and BGOA-V and BGOA-M was proposed. BGOA-S and BGOA-V methods use S-shaped (Kennedy & Eberhart, 1997) and V-shaped (Rashedi et al., 2010) transfer functions to convert GOA's solutions to binary. On the other hand, BGOA-M updates the position of the current solution based on the position of best solution so far. It also uses a mutation operator to enhance the exploration and avoid early convergence at the appropriate mutation rate. The used mutation rate is linearly reduced from 0.9 to 0 during the iterations. Like Key 13 The maximum number of iterations (100) Superiority of this method over the BGWO, GA, Scatter Search algorithm (SSA), ACO, PSO, Correlation-based FS (CFS), FCBF, F-Score, IG, and ReliefF on 25 UCI datasets.	1) The changing mechanism of BGOA-M will encourage more exploration and the mutation operator significantly affects the performance of BGOA. 2) The BGOA-M is superior to two other techniques.	16
(Hichem et al., 2019)	Novel Binary GOA (NBGOA) normalises the distance between grasshoppers based on Hamming distance and then updates the position of grasshoppers. Each solution in NBGOA are defined similar to Key 3. Like Key 13 The maximum number of iterations (100) Superiority of this method over the BGOA-M, Binary Dragonfly Optimisation (BDO), BGOA, BGWOA and Naïve Bayes PSO (NBPSO) on 20 datasets from UCI.	The NBGOA improved the exploration and computational time by the binary initialisation of population.	17

(*Continued*)

Table 5. (Continued).

Publication	Methodology Fitness function Stopping Criteria Experimental results	Findings	Key NO.
(H. T. Ibrahim et al., 2019)	Because the determination of SVM's kernel factors affects the classification performance, GOA+SVM was proposed which improves the SVM factors and selects the best features subset using GOA. The features vector is similar to key 3. The SVM classification accuracy The maximum number of iterations Superiority of GOA+SVM over the Multi-Verse Optimizer (MVO), GA, PSO, Bat Algorithm (BA), and Firefly Algorithm (FA) on Iraqi cancer datasets from 2010 to 2012 and University of California Irvine datasets.	1) The best accuracy was obtained when increasing the number of search agents. 2) The convergence of this method was rapid for large real datasets.	18
(Zakeri & Hokmabadi, 2019)	GOA Feature Selection (GOFS) uses a mathematical model that is inspired by the repulsion and attraction forces between grasshoppers to explore the features space and to exploit the good features subset. This method uses an adaptive reducing parameter for shrinking comfort, repulsion, and attraction areas which in turn, balances between exploration and exploitation. Also, a probability-based distribution factor and rounding operation are used for substituting the duplicate features and Out-of-range indices, respectively. The SVM or KNN classification error rate The maximum number of iterations (100) Superiority of this method over the BGA, PSO, ACO, SA, and Differential Evolution Feature Selection (DEFS) on ten UCI datasets.	The feature goodness factor helps GOFS to search promising features and find the global solution	19

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

Publication	Methodology Fitness function Stopping Criteria Experimental results	Findings	Key NO.
(Gu et al., 2018)	The Competitive Swarm Optimizer (CSO) works based on two techniques: 1) Dividing randomly the population into two groups, carrying out pairwise competitions between the particles from each group, passing directly winner particle to the next iteration and updating position and velocity of loser particle by learning from the winner particle; 2) Using an archive technique to record the fitness values of all previous feature subsets to reduce the number of fitness calculations. Each solution is encoded similar to key 3. The average error rates of KNN classifier The maximum number of iterations (200) Applying the method to six UCI datasets, and superiority of this method over the conventional PCA-based method and some versions of PSO on high-dimensional datasets.	Using the simple archive technique significantly reduces the search time.	20
(Jain et al., 2018)	Correlation-based Feature Selection-Improved Binary Particle Swarm Optimisation Naïve Bayes (CFS-IBPSO-NB) has two phases: 1) Predictive Gene Pre-Filtering (PGPF) phase in which, the missing values are replaced with mean values, the dataset is standardised and the data dimensions are reduced by using CFS for eliminating the irrelevant and redundant genes; 2) Gene Optimisation and Cancer Classification (GOCC) phase that uses IBPSO-NB wrapper method and 10-fold cross-validation to select the best subset of genes. In addition, each particle's position changes to binary value by using a sigmoid function. The NB classification error The maximum number of iterations (100) Superiority of this method over SVM, Random Forest (RF), FCBF, BPSO, PSO-DT, Markov Blanket-Embedded Genetic Algorithm (MBEGA), and Taguchi Chaotic Binary Particle Swarm Optimisation (TCBPSO) on 11 cancer microarray datasets.	1) This method is a good tool for the DNA microarray analysis and early prognosis of the Cancer by selecting the small number of high-relevant genes. 2) Complexity of the method is better than the compared methods.	21

(*Continued*)

Table 6. (Continued).

Publication	Methodology / Fitness function / Stopping Criteria / Experimental results	Findings	Key NO.
(Rajamohana & Umamaheswari, 2018)	Hybrid Improved Binary PSO and Shuffled Frog Leaping Algorithm (Hybrid iBPSO-SFLA) method uses iBPSO to achieve the best feature subset and then, the SFLA (Eusuff et al., 2006) to obtain the optimal feature subset. Each particle position in iBPSO is similar to Key 3. Since the inertia weight plays a critical role in the exploration and exploitation processes, iBPSO uses the combination of Linearly Decreasing Inertia Weight (LDIW) with the convergence coefficient to calculate the particles' velocity. The NB classification accuracy Convergence criteria or the maximum number of iterations (500) Applying to a dataset developed (Ott et al., 2011) and evaluating by NB, KNN and SVM classifiers, indicated the best classification performance was achieved with NB classifier.	Using Inertia weight in the velocity of iBPSO is effective for enhancing the search speed and balancing between exploration and exploitation.	22
(Engelbrecht et al., 2019)	This wrapper feature selection method uses Set-Based PSO (SBPSO) (Langeveld & Engelbrecht, 2012) and KNN classifier. SBPSO defines a particle's position and velocity as mathematical sets. The position and velocity of each particle in SBPSO are defined based a set of elements from the universe of problem and a set of addition or deletion operation pairs, respectively. The best position represents the optimal solution of SBPSO. 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. Superiority of this method over the BPSO, Catfish Binary PSO (CFBPSO), and Probability Binary PSO (PBPSO) on 30 UCI datasets.	The performance of KNN classifier in SBPSO is better than the Gaussian NB and DT J48.	23
(Chen et al., 2019)	A Hybrid Particle Swarm Optimisation with a Spiral-Shaped Mechanism (HPSO-SSM) is a unsupervised feature selection which uses (1) A logistic map sequence (G. -G. Wang et al., 2014) 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 to balance exploration and exploitation. (3) A spiral-shaped mechanism (Mirjalili & Lewis, 2016) to enhance the solution quality. Therefore, the position of each	1) Using the logistic map sequence improves diversity of population and escapes from local optima. 2) Two dynamic corrections help the	24

(*Continued*)

Table 6. (Continued).

Publication	Methodology / Fitness function / Stopping Criteria / Experimental results	Findings	Key NO.
	particle is updated with the probability of 80% by the proposed position update formula or with the probability of 20% by the spiral-shaped formula.	particles to find a new area.	
	Like Key 13		
	number of iterations (100)		
	Superiority of this method over the Whale Optimisation Algorithm (WOA), Biogeography-Based Optimisation (BBO), Kill Herd algorithm (KH), ABC, IG, Spectrum, F-Score, ReliefF, CFS, and FCBF on 20 UCI datasets.		
(Kumar & Bharti, 2019)	The Modified-BPSO method was proposed for enhancing clustering performance. The best swarm is selected with the help of Silhouette index (SI) (Kaufman & Rousseeuw, 2009). Also, the goodness of clusters is calculated by using different cluster validation indices such as Dunn index (DI) and Davies-Bouldin Index (DBI).	The higher SI value, the more relevant features are selected, and eventually clustering accuracy is enhanced.	25
	The SI		
	The maximum number of iterations (150)		
	Superiority of this method over the BPSO and Binary Moth Flame Optimisation (BMFOA) on six real datasets from UCI repository.		
(Y. Zhang et al., 2019)	Filter-based Bare-bone PSO (FBPSO) is unsupervised feature selection that uses four strategies: 1) an Average Mutual Information(AMI)-based space reduction strategy (Y. Wang et al., 2017) to remove irrelevant and weakly-relevant features, 2) a feature redundancy- based local filter search strategy to improve the exploitation by deleting the redundant features and adding the missing important features, and 3) a similarity-based assessment function and a parameter-free update strategy to get the high performance. 4) a probability-based encoding strategy to encode a particle. If the selecting probability of a feature is greater than the threshold value of 0.5, this feature is selected; otherwise, it is left unselected.	The used strategies in this method, help to remove irrelevant and redundant features quickly, and also enhance convergence speed and exploitation.	26
	$fit_s - fit_{dis}$, where fit_{dis} and fit_s are dissimilarity of the selected features and the similarity of unselected features.		
	The maximum number of iterations (100)		
	The results obtained from applying this method to two biological datasets from the UCI repository, two image datasets, and two text datasets from the ASU repository confirmed the effectiveness of FBPSO as an unsupervised feature selection method.		

(*Continued*)

Table 6. (Continued).

Publication	Methodology / Fitness function / Stopping Criteria / Experimental results	Findings	Key NO.
(Tadist et al., 2021)	The Spark Distributed PSO (SDPSO) was proposed for cancer prognosis purposes by combination an integrative BPSO feature selection with PSO algorithm. SDPSO has three main phases: Cleaning datasets, the BPSO-based wrapper method, and a hybrid PSO-KMeans algorithm to achieve a better performance. The proposed BPSO uses Mutual information (MI) for evaluating the feature relevancy and redundancy in selected subset.	1) The hybrid PSO-KMeans accelerates the convergence speed and accuracy. 2) The runtime of SDPSO is low.	27
	Like Key 13		
	The maximum number of iterations (15)		
	SDPSO obtained the best result compared to K-Means, GA, DBSCAN, and hybrid PSO-GA on five genomic datasets in terms of average purity and F-measure.		

suitable for small and medium size datasets. The efficiency of these methods decrease as the number of features increases. The number of selected features by the HPSO-SSM method is still high for many data sets. Also, due to the logistic map sequence time and the calculation time for the dynamic correction factors, the time computational complexity of this method is slightly high. The FBPSO is a good unsupervised feature selection method. But, this method should be developed to support biomedical datasets in disease diagnosis and image recognition area. The SDPSO depends on user-defined parameters that affect final optimal results.

Crow Search Algorithm-based feature selection

Crow Search Algorithm (CSA) is an optimisation algorithm proposed by (Askarzadeh, 2016), which mimics the crows' search behaviours to hide their food. Due to high performance, CSA can solve the feature selection problem. Table 7 demonstrates the primary details taken from the recently-proposed CSA-based feature selection algorithms.

As a conclusion can be said, the performance of CCSA decreases with increasing number of features. And the CFCSA should be developed to support various other areas and reduce runtime.

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

Publication	Methodology Fitness function Stopping Criteria Experimental results	Findings	Key NO.
(Sayed, Hassanien, et al., 2019)	A Chaotic Crow Search Algorithm (CCSA) uses chaotic variables instead of the random variables in updating the crow position. Each solution of CCSA, have a binary form like Key 3. $Accuracy + \alpha(1 - \lvert S\rvert/\lvert N\rvert)$, $\lvert S\rvert$ and $\lvert N\rvert$ are the number of selected features and the original number of features, respectively. $\alpha = 0.8$. The maximum number of iterations (5) or reaching the best solution. Superiority of this method over the PSO, ABC, Flower Pollination Algorithm (FPA), Moth-Flame Optimisation (MFO), GWO, WOA, Sine Cosine Algorithm (SCA), Chicken Swarm Optimisation (CSO) and CSA on 20 medical/biology and business UCI datasets.	The sine chaos map is a desirable map to dramatically enhance the classification accuracy, stability and convergence speed and to reduce the number of features.	28
(Anter & M, 2020)	hybrid CFCSA method was proposed by combination of CSA with Chaos Theory (CF) and Fuzzy C-Means algorithm (FCM) to solve the feature selection problem in medical diagnosis. CFCSA uses the global optimisation technique to solve the local minima problem, the chaos theory to improve the low convergence of CSA and the FCM (Bezdek et al., 1984) to control uncertainty and evaluate crows in high-dimensional spaces. The FCM cost function The maximum number of iterations (20) Superiority of this method over the BCSA, CALO, BALO, and BA on 20 medical datasets.	The classification accuracy of CFCSA is significantly enhanced by using Singer, Gauss, and Tent maps.	29

Gray Wolf Optimization-based feature selection

GWO is a meta-heuristic algorithm that mimics the social behaviour and hierarchy of grey wolves while hunting and it was first proposed by (Mirjalili et al., 2014). This algorithm can solve large-scale problems. The hunting process of grey wolves has three stages: tracking and approaching, pursuing and encircling, and attacking. In recent years, various methods based on this algorithm have been proposed to solve the feature selection problem, such as bGWO (Emary et al., 2016), GWO-ANN (Chowdhury et al., 2020). Table 8 demonstrates the primary details taken from the some of these methods.

The efficiency and runtime of LFGWO and MGWO on the other types of data, GWOCSA on large datasets and HSGW on big datasets have not been investigated. The performance and stability of BGWOPSO by using two other classifiers such as SVM and ANN could be compared and evaluated. The time consumption of TMGWO is higher than the other compared methods.

Gravitational Search Algorithm-based feature selection

Gravitational Search Algorithm (GSA) was presented by (Rashedi et al., 2009) and inspired by Newton's law of gravity. Each object perceives its location and the position of other objects by gravity, and each object affects other objects in proportion to its mass and distance from the other objects. Table 9 demonstrates the primary details taken from recent papers on GSA-based feature selection.

HGSA is suitable for small datasets. This method can be developed to support large data sets in other domains.

Butterfly Optimization Algorithm-based feature selection

Butterfly Optimisation Algorithm (BOA) was proposed by (Arora & Singh, 2017). This method is inspired by the searching behaviours of butterflies. Each butterfly produces scent that can be felt by different neighbouring butterflies and a general system of social learning is formed. BOA has a high efficiency and convergence rate due to the use of random and decentralised steps. Table 10 demonstrates the primary details taken from recently-proposed BOA-based feature selection algorithms.

According to the studies, it can be said that the bBOA method did not consider redundancy. On the other hand, although the EITBBOA considers redundancy, it has a higher execution time than the compared methods. So parallelism can help solve this problem. This method is also suitable for medical datasets. The development of this method is necessary to support other types of data.

Dragonfly Algorithm-based feature selection

Dragonfly Algorithm (DA) is a population-based optimisation proposed by (Mirjalili, 2016) and is inspired by dragonfly hunting and migrating behaviour. DA Operators provide two basic concepts: diversification and intensification, which are resulted from the static swarming (Hunting strategy) and dynamic swarming (migration strategy) activities, respectively. Table 11 demonstrates the primary details taken from the recent studies carried out in the DA-based feature selection.

Studies showed high performance of the BDA and CDA methods on large scale medical datasets. These methods can be extended to support other areas.

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

Publication	Methodology Fitness function Stopping Criteria Experimental results	Findings	Key NO.
(Pathak et al., 2019)	The Levy Flight-based GWO (LFGWO) method was proposed for image steganalysis. LFGWO uses the LF (Chechkin et al., 2008) to increase the step size of a search agent, which enhances the ability of the exploration and exploitation of GWO. For calculating the jump size, LF uses the Levy probability distribution function (function of power-law). This method has three phases: extracting features from the input images by using the Subtractive Pixel Adjacency Matrix (SPAM) (Pevny et al., 2010) or AlexNet, a Convolution Neural Network (CNN) (Krizhevsky et al., 2017), reducing the dimension of features by applying the LFGWO-based feature selection, and applying the selected features to a classifier. The position of each grey wolf in LFGWO is initialised with a random vector of numbers in range 0 to 1. Each element of this vector is set to 0, if its value is less than the threshold 0.9; otherwise, it is set to 1. The SVM accuracy Not mentioned Superiority of this method over the Fast Evolutionary Programing (FEP), GWO, PSO, and GSA on the BOSS base ver. 1.01 dataset.	1) Used of LF in LFGWO makes a good balancing between exploration and exploitation, that helps to find the accurate solution. 2) The LFGWO can also remove the irrelevant and redundant features while keeps the classification accuracy in high.	30
(Al-Tashi et al., 2019)	binary version of PSOGWO (Singh & Singh, 2017), called BGWOPSO, uses the KNN classifier with the Euclidean separation matrix to find the best solution. A sigmoid function is also used to convert the position of wolves to the binary. Each solution is illustrated in binary form similar to Key 3. Like Key 13 The maximum number of iterations (100) Superiority of this method over the Whale Optimisation Algorithm with Simulated Annealing (WOASAT-2), BGWO, BPSO, and BGA on 18 UCI datasets.	BGWOPSO performs a good balancing between the exploration and exploitation based on updated position of the three best solutions.	31
(P. Sharma et al., 2019)	To help detect Parkinson's disease symptoms at a premature stage, a modified version of GWO, called MGWO was proposed. This method has been expanded for multiple features selection. The position of each wolf is updated by a sigmoid function based on the position of the alpha, beta and delta wolves. Like Key 27, $\alpha = 0.7$ The maximum number of iterations (1000) Superiority of this method over the Optimised Cuttlefish Algorithm (OCFA) on voice, handwriting (spiral and meander), and speech datasets with RF, KNN and DT classifiers. And, superiority of RF in	The weight factor establishes a balancing between the importance summation of the selected features, the importance of the candidate feature, and the size of subset.	32

(*Continued*)

Table 8. (Continued).

Publication	Methodology Fitness function Stopping Criteria Experimental results	Findings	Key NO.
	the MGWO over the DT and KNN.		
(Tu et al., 2019)	Multi-strategy Ensemble GWO (MEGWO) has three different search strategies for updating solutions: the enhanced global-best lead strategy to enhance local search ability of GWO, the adaptable cooperative hunting strategy to increase population diversity and the ability of global search, and the disperse foraging strategy to balance between exploitation and exploration. For these purposes, the way of Global-best Harmony Search (GHS) (Omran & Mahdavi, 2008) and a modified differential mutation operator are used in the first strategy for exploring the excellent experience of the alpha wolf and Lack of search bias towards exploitation. One-dimensional update operation inspired by ABC and total-dimensional update operation are alternatively used in second strategy. Also, a self-adjusting parameter is used in third strategy to explore a promising area by some agents. Like Key 13 The maximum number of iterations (2500) Superiority of MEGWO based on different CEC2014 functions over the Adaptive Cuckoo Search (ACS), Comprehensive Learning PSO with Local Optima Topology (CLPSO- LOT), Differential Evolution with Biogeography-Based Optimisation (DE/BBO), Biogeographic Harmony Search (BHS), Sinusoidal Differential Evolution (SinDE), Multi-strategy Ensemble ABC and WOA on 12 UCI datasets.	MEGWO can solve real-world optimisation problems with high efficiency and reliability.	33
(Arora et al., 2019)	Gray Wolf Optimisation Crow Search Algorithm (GWOCSA) combines GWO and CSA to get the global optima by generation promising candidate solutions. GWOCSA uses an adaptive balance probability to get acceleration throughout the early steps of optimisation process and to enhance exploitation from promising solutions in the later steps and a nonlinear decreasing control parameter to enhance the search capacity. Like Key 13 The maximum number of iterations (100) Superiority of GWOCSA over the ALO, CSA, FPA, PSO, Spotted Hyena Optimisation (SHO), GA, Adaptive GWO (AGWO), GWO and EGWO on 23 benchmark functions.	1) GWOCSA can solve complex real- world problems by combining the GWO's exploitation and the CSO ability of avoiding local optima. 2) GWOCSA has a complete superiority over the GWO, AGWO and EGWO.	34

(*Continued*)

Table 8. (Continued).

Publication	Methodology Fitness function Stopping Criteria Experimental results	Findings	Key NO.
(Mafarja, Qasem, et al., 2020)	Three hybrid methods, Serial Grey-Whale (HSGW), Random Switching Grey-Whale (RSGW), and Adaptive Switching Grey-Whale Optimisation(ASGWO) were proposed by combination of GWO and WOA. HSGW uses a two-phases updating, in which a solution is placed around the best three solutions of GWO, at first and then search process extends to other promising zones by WOA operators. WOA works on the same population used by GWO and the leader wolf (alpha) is updated. In RSGW, the proposed procedure switches between GWO and WOA based on a generated random number between 0 and 1. The representation of each solution in these methods is similar to key 3. Like Key 13 The maximum number of iterations (100) Superiority of this method over the GA, BPSO, BGSA, and BGOA on 18 datasets of UCI repository.	1) HSGW achieved better result compared to RSGW and ASGW in most of the datasets used. 2) ASGWO has low computational time	35
(Abdel-Basset et al., 2020)	Two-phase Mutation GWO (TMGWO) uses 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. The number of selected features is decrease by employing the first phase of mutation while keeping high classification accuracy and the more informative features is added by applying the second phase of mutation. This method archives high-quality solutions by using KNN classifier with Euclidean distance as an evaluator. Each possible solution in this method is a binary vector similar to Key 3. Like Key 13 The maximum number of iterations (30) Superiority of this method over the Flower Algorithm (FA), PSO, Multi-Verse Optimizer algorithm (MVO), WOA, BCSA, Non-Linear Particle Swarm Optimisation algorithm (NLPSO), and BA on 35 datasets from UCI repository.	The proposed mutation operator effectively helps GWO to find the best subset.	36

Table 9. Details of methodology and findings of the GSA-based feature selection.

Publication	Methodology Fitness function Stopping Criteria Experimental results	Findings	Key NO.
(Taradeh et al., 2019)	Hybrid GSA-based feature selection uses four mechanisms: 1) a logarithmic decreasing function instead of the linear decreasing function for gradually updating the gravitational constant, 2) GA's crossover operator- based mechanism for generating new solutions and increasing the GSA exploratory powers, 3) A counting mechanism of the number of times that the best solution obtained so far has not been improved, for checking whether the algorithm is stuck in local optima or not. If yes, the GA's mutation operator is applied to the gbest, and 4) the V-shaped function to transform the continuous space to the binary. Like Key 13 The maximum number of iterations (200) Superiority of this method over the GA, PSO, GWO, GSA, and MFO showed that HGSA on 18 well-known UCI datasets.	1) HGSA avoids the worst misleading agents and Gradually improves the quality of the population by crossover operator. 2) The quality of solutions and exploitation of HGSA is enhanced by the fitness-based mutation operator.	37

Salp Swarm Algorithm-based feature selection

Salp Swarm Algorithm (SSA) is a population-based optimisation algorithms proposed by (Mirjalili et al., 2017). SSA mimics the behaviour of salps swarm when searching for food in the oceans. The position of salps is defined based on the number of given problem's variables.

Table 12 demonstrates the primary details of the chosen SSA-based recent papers.

Studies show that the TCSSA and CSSA methods have a good stability. Also, efficiency and execution time of these six methods not been investigated for other data types and larger datasets. It seems that these methods should be developed to support big data and other areas such as image processing, signal processing, etc.

Coral Reefs Optimization-based feature selection

The Coral Reef Optimisation (CRO) algorithm is a random search optimisation algorithm proposed by (Salcedo-Sanz et al., 2014). The CRO algorithm has two phases: 1) reef formation phase that is initialised by a reef of solutions to an optimisation problem, and 2) coral reproduction that searches for optimal solutions by Broadcast spawning, brooding and budding operators. New individuals (larvae) compete with the others for space in the reef. And finally, the individual with a better fitness will occupy the reef. Table 13 demonstrates the primary details taken from recent papers published on the CRO-based feature selection.

It seems that although the efficiency of disease diagnosis and execution time of BCRO have acceptable quality, the application of this method is limited to medical data. So, BCRO can be improved by integrating with other search strategies or optimisation algorithms to support the other data and applications.

Table 10. Details of methodology and findings of the BOA-based feature selection algorithms.

Publication	Methodology Fitness function Stopping Criteria Experimental results	Findings	Key NO.
(Arora & Anand, 2019)	Two versions of the binary BOA (S-bBOA and V-bBOA) were proposed that use the Sigmoid transfer function $(1/(1+e^{-F_i^k(t)}))$ and the V-shaped transfer function $(\|\sqrt{\pi}/2\ {}_0^{(\sqrt{\pi}/2)F_i^k(t)}\ e^{-t^2}dt\|)$. Like Key 13 The maximum number of iterations (100) Superiority of these methods over the DA, ALO, GA, PSO, GWO, SSA, Sine-Cosine Algorithm (SCA), Brain Storm Optimisation (BSO) and WOA on 21 UCI datasets.	The S-bBOA is better than the V-bBOA in converge, effectively search, finding the accurate best solution and enhancing the performance of BOA.	38
(Sadeghian et al., 2021)	Ensemble Information Theory binary BOA (EIT-bBOA), has three phases: The Minimal Redundancy-Maximal New Classification Information (MR-MNCI) method (Gao et al., 2018) in the first phase to select 20% of the relevant and non-redundant features, The Information Gain bBOA (IG-bBOA) with a three-purpose fitness function in the second phase to find an optimised feature subset, and finally, an ensemble similarity-based method and selecting 30 best features. Each solution of IG-bBOA is a binary vector similar to Key 3. $\alpha Accuracy + \beta(\|N-S\|/\|N\|) + \delta Mean(I(X_k;Y))$, where $\|N\|$ is the number of original features, $\alpha = 0.99$, $\beta = 0.001$, and $\delta = 0.009$. The maximum number of iterations (100) Superiority of this method over the bGWO, bWOA, and bCSA on six UCI datasets.	1) The use of the mean of IG makes more diversity to reaches the optimal solution with the high accuracy. 2) EIT-bBOA is mostly used for feature selection in medical datasets. 3) The EIT-bBOA has acceptable stability.	39

Spotted Hyena Optimization-based feature selection

The Spotted Hyena Optimisation (SHO) is a swarm-based algorithm proposed by (Dhiman & Kumar, 2017) and is inspired by the spotted hyenas social hierarchy and hunting behaviours. SHO have three basic phases: searching for prey, encircling, and attacking prey. Finding the best solution in a short time and self-adaptation are the advantages of SHO. Table 14 demonstrates the primary details taken from the recently-proposed SHO-based feature selection algorithms.

Cuckoo Search-based feature selection

Cuckoo Search (CS) is an optimisation algorithm proposed by (Yang & Deb, 2009) that was developed for non-linear and continuous optimisation problems. This method was inspired by the interesting cuckoo lifestyle and spawning. Table 15 demonstrates the primary details taken from recent studies conducted into the CS-based feature selection.

The AGCS-PBAMNN works well for abnormal features and the MCSRS has a good performance in selecting fewer features, deleting irrelevant and redundant features, maximising classification accuracy and high stability. Nevertheless, the effectiveness of AGCS-PBAMNN and MCSRS has not been investigated for the very large datasets in other domains and applications.

Table 11. Details of methodology and findings of the DA-based feature selection algorithms.

Publication	Methodology / Fitness function / Stopping Criteria / Experimental results	Findings	Key NO.
(Sayed, Tharwat, et al., 2019)	Chaotic DA (CDA) uses the Synthetic Minority Over-Sampling Technique (SMOTE) to refine the unbalanced datasets in the first phase, CDA to select the most distinctive features in the second phase, and SVM classifier to evaluate the selected subset in the last phase. Like Key 27. Just, w is set to 0.9 in this method. The maximum number of iterations (50) or finding the best solution. Superiority of this method over the DA, PSO, CSO, GWO, and ABC on the extracted dataset from the Drug bank.	1) Combining chaotic map with DA makes a fast, robust and efficient feature selection model. 2) The Gauss map enhances performance of DA more than the other chaotic maps.	40
(Mafarja, Heidari, et al., 2020)	BDA uses DA to find the best subset and KNN to evaluate the subset. Each solution in BDA is represented by a binary vector similar to Key 3. Furthermore, a V-shaped function ($\left\|v_d^i(t)\right\| / \left(1 + v_d^i(t)^2\right)^{1/2}$) is used to transfer the continuous positions to binary space. Like Key 13 Not mentioned Superiority of this method over the BGWO, BGSA, BBA, BGA, and BPSO on 9 large medical datasets	BDA is suitable for high dimensional medical datasets.	41

Whale Optimization Algorithm-based feature selection

The Whale Optimisation Algorithm (WOA) is a population-based algorithm proposed by (Mirjalili & Lewis, 2016) based on the encircling and hunting behaviours of humpback whales. WOA has three phases: 1) encircling prey that 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). Table 16 presents the primary details of the chosen papers recently published on WOA-based algorithms.

Firefly Algorithm (FA) (Yang, 2010) is a swarm intelligence-based optimisation methods, that is inspired by the brightness of fireflies in nature. This algorithm is used in engineering fields, nonlinear multi-quality optimisation problems as well as NP-Hard problems. In the FA, a kind of random search is employed to reach a set of solutions. One of the advantages of this search mechanism is that it avoids falling into the local optimum trap. Table 17 demonstrates the primary details of the selected recent papers published on applying the FA-based algorithms to feature selection problems.

The BFA+Penalty+RF is a good stability, but the application of this method is limited to medical domain and cancer diagnosis. Also, the prop. FA method is limited to small datasets without missing values, and this method should be developed to work on large and complex datasets. Due to the use of an estimate process for the optimal fitness values, FGACO has a high performance and selects at least the number of features at the minimum time.

Fruit Fly Optimization Algorithm-based feature selection

The Fruit Fly Optimisation Algorithm (FFOA) is an evolutionary optimisation method that was first introduced by (W. -T. Pan, 2012) and is inspired by fruit flies food searching behaviours. FFOA has two phases; The fruit flies fly towards food place by using their smell capability, at the first phase. In the second phase, they get nearer by using their vision power. Simple structure, few parameters to set, solving problems with high speed, and high stability are the advantages of this algorithm. Table 18

Table 12. Details of methodology and findings of the SSA-based feature selection algorithms.

Publication	Methodology	Fitness function	Stopping Criteria	Experimental results	Findings	Key NO.
(Faris et al., 2018)	Two binary SSA-based feature selection methods were proposed. In the first method, four S-shaped and four V-shaped transfer functions are employed to convert the continuous space into the binary. The second method uses a crossover operator instead of the average operator in position updating mechanism to increase the SSA exploratory behaviour. The crossover operator switches between two input vectors with the same probability.	Like Key13	Not mentioned	Superiority of this method over the BGWO, BGSA, BBA, BPSO, and GA on 22 UCI datasets.	1) The crossover operator enhances the performance of BSSA by promoting exploration. 2) The BSSA with transfer function $1/\left(1+e^{(-x/2)}\right)$ and crossover achieved the better results than the other.	42
(Aljarah et al., 2018)	Termite Colony SSA (TCSSA) is a binary SSA with synchronous updating rules and a new TC-based leadership structure. The salp chain is divided into several sub-chains; then, salps perform a different strategy in each sub-chain to adaptively change their positions. Based on the adopted strategies, three distinct versions of SSA (called TCSSA1, TCSSA2, and TCSSA3) were presented. For updating controller variable in the position updating formula, the TCSSA1 utilises $(-2.05t/T)+2.55$ in Sub-chains 1 and 2, and $(-2t^3/t^3)+2.5$ in Sub-chains 3 and 4. The TCSSA2 uses $2.5+2(t/T)^2-2(2t/T)$ in Sub-chain 1, $0.5+2\exp\left[-(4t/T)^2\right]$ in Sub-chain 2, $(-2t^3/T^3)+2$ in Sub-chain 3, and $2.5-(2\log(t)/\log(T))$ in Sub-chain 4. The TCSSA3 algorithm utilises $1.95-2t^{1/3}/T^{1/3}$ in Sub-chains 1 and 2, and $(-2t^3/t^3)+2.5$ in Sub-chains 3 and 4.	Like Key13	The maximum number of iterations (100)	Superiority of this method over the PSO, BGWO, BGSA, and BBA on 20 UCI repository datasets.	1) Selecting half of the salps as leaders is the best leadership structure, which improves the accuracy, exploration and exploitation of SSA. 2) The updating strategy $1\cdot 95-2t^{1/3}/T^{1/3}$for Sub-chains 1 and 2, and $(-2t^3/T^3)+2\cdot 5$ for Sub-chains 3 and 4 in the majority of datasets is the best updating strategy.	43
(Hegazy et al., 2019)	The Chaotic SSA (CSSA) uses a chaotic map such as logistic, piecewise, singer, sinusoidal, and tent to increase the SSA performance in the optimisation process. Each solution in CSSA is a binary vector similar to Key 29 with threshold 0.5.	Like Key13. Just, $\alpha=0.9999$.	The maximum number of iterations (50)	Superiority of this method (especially with the tent map) over the standard SSA, PSO, and GA on 27 UCI datasets.	The created chaotic sequence by Chaotic maps, updates the salp position, improves the optimal solution and speeds up convergence.	44

(*Continued*)

Table 12. (Continued).

Publication	Methodology / Fitness function / Stopping Criteria / Experimental results	Findings	Key NO.
(R. A. Ibrahim et al., 2019)	SSAPSO was proposed by combining SSA with PSO to enhance the exploration and exploitation power. In this method, the population is updated by using SSA or PSO based on its probability value ($fitness_i / \sum_{i=1}^{N} fitness_i$). If the probability value is less than or equal 0.5, then the PSO is used; otherwise, the SSA is used. Each solution has a representation similar to Key 3.	The quality of the SSA in searching and creating diversity in the population is enhanced by the capabilities of the PSO. Thus, the convergence is speed up.	45
	Like Key 13		
	The maximum number of iterations (200)		
	Superiority of this method over the GA, PSO, BAT, and SSA on 10 UCI datasets.		
(Hegazy et al., 2020)	Improved Salp Swarm Algorithm (ISSA) uses the inertia weight to modify the current best solution and KNN classifier to evaluate solutions. The representation of solutions and transfer function are similar to the Key 43	The inertia weight enhances the convergence speed and reliability of SSA, avoids a vast number of local solutions and achieves an exact appreciation of the optimal solution.	46
	Like Key 13		
	The maximum number of iterations (50)		
	Superiority of this method over the basic SSA, GA, PSO, ALO, and GWO on 23 UCI datasets.		
(Tubishat et al., 2021)	Dynamic SSA (DSSA) uses two enhancements: 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.	1) The new position formula enhances the diversity of solutions. 2) The use of LSA reduces the computational time and improves the best current solution.	47
	The KNN classification error		
	The maximum number of iterations (10)		
	Superiority of this method over the SSA, PSO, GA, ALO, and GOA on 20 UCI datasets and 3 Hadith datasets.		

Table 13. Details of methodology and findings of the CRO-based feature selection algorithms.

Publication	Methodology / Fitness function / Stopping Criteria / Experimental results	Findings	Key NO.
(Yan et al., 2019)	Binary Coral Reef Optimisation Simulated Annealing Tournament (BCROSAT), was proposed by combination the BCRO with TS for updating the worst solution and SA for updating the best solution with the higher fitness. This method uses KNN classifier to evaluate solutions. Each generated larva (solution) at the first phase of this method is a binary vector similar to Key 3. Also, The broadcast spawning is performed by a crossover operator.	1) Due to the high local search power of SA, the search performance of the CRO improved. 2) This method is a good tool for pre-processing on high dimensional biomedical data.	48
	Like Key 13		
	Not mentioned		
	Superiority of BCROSAT over the Whale Optimisation Algorithm Simulated Annealing Tournament (WOASAT), Improved Genetic Algorithm (IGA), Modified Binary PSO (MBPSO),, and Improved SFLA (ISFLA) on 13 high-dimensional biomedical datasets.		

demonstrates the primary details of the chosen FFOA-based papers recently published on the feature selection problem.

Despite its high accuracy, BIFFOA is still weak in decreasing size of selected features subset. Low stability in high dimensional datasets and falling into local optimal are the limitations of FOA method.

Brain Storm Optimization-based feature selection

The Brain Storm Optimisation (BSO) algorithm proposed by (Shi, 2011) is a swarm intelligence method based on the human beings' brainstorming processes. BSO has three steps: clustering the

Table 14. Details of methodology and findings of the SHO-based feature selection algorithms.

Publication	Methodology / Fitness function / Stopping Criteria / Experimental results	Findings	Key NO.
(Jia et al., 2019)	Two different hybrid SHO-based feature selection models were proposed. First model (called SHOSA-1) uses SA as an operator of SHO to gain the best solution and reposition the main solution around a randomly-selected solution and the known best solution. The second model (called SHOSA-2) uses SA to improve the final obtained solution from SHO.	SHOSA-1 has an excellent performance; however, it is not suitable for high dimensional datasets because of a long-running time.	49
	Like Key 13		
	The maximum number of iterations (30)		
	Superiority of the method over the ACO, CS, FPA, PSO, SCA, and SSA on 20 UCI datasets. And superiority of SHOSA-1 over the native algorithm and SHOSA-2.		

Table 15. Details of methodology and findings of the CS-based feature selection algorithms.

Publication	Methodology / Fitness function / Stopping Criteria / Experimental results	Findings	Key NO.
(Abdel Aziz & Hassanien, 2018)	A Modified CS with Rough Sets (MCSRS) was proposed for high-dimensional datasets. This method uses LF to update the position of cuckoo. The crossover between cuckoos is performed as: (1) For all the top elite eggs x_i randomly picks elite egg x_j, and (2) The new solution x_k is generated by computing the distance between x_i and x_j. The distance is computed in two ways; 1) If the fitness values of x_i and x_j are equal, then the distance is $\lvert x_i - x_j \rvert / 2$. 2) The inverse of the golden ratio is used $(\lvert x_i - x_j \rvert / ((1+\sqrt{5})/2))$. Like Key 13 The maximum number of iterations Superiority of MCSRS over the GARS, PSORS, GRSARS, MCSRS, IHSRS, and FARS on 7 UCI datasets.	With the use of Rough Set for evaluation of the importance of features and Levy flights as the search method and applying dynamic step, the performance of MCSRS is enhanced.	50
(Jayaraman & Sultana, 2019)	AGCS-PBAMNN was proposed for the heart disease classification systems. This method combines the artificial gravitational cuckoo search (AGCS) (Yang & Deb, 2009) as a feature selection method with the particle bee optimised associative memory neural network (Bryll et al., 2003) as a feature classifier. (Kumar & Bharti, 2019) The maximum number of iterations Superiority of the method over the Hybrid GA with a new Local Search algorithm (HGLSA), Hybrid PSO with Wrapper-Filter (HPSOWF), and Hybrid ACO (HACO) on UCI Heart Disease Dataset.	Because the features selected by AGCS are more relevant to practical solutions,heart disease is successfully classified.	51

Table 16. Details of methodology and findings of the WOA-based feature selection algorithms.

Publication	Methodology / Fitness function / Stopping Criteria / Experimental results	Findings	Key NO.
(Zheng et al., 2018)	Maximum Pearson Maximum Distance Improved WOA (MPMDIWOA) was proposed by combination of the Maximum Pearson Maximum Distance (MPMD) with Improved WOA (IWOA). In addition, Maximum Value Without Change (MVWC) and threshold changes is used for adjusting the sequence and frequency of run. The MPMD method uses the maximum Pearson correlation coefficient for measuring the relevance between class labels and features, and the Maximum Correlation distance for measuring the redundancy between features. Moreover, IWOA saves the first three optimal solutions and generates a new position based on voting between these three position. The SVM classification accuracy The maximum number of iterations (100) Superiority of the method over the PSO, GA, and BBA, and the hybrid MPMDMVO on ten UCI datasets.	Use of the voting method in IWOA makes more exploration and finding global optimal values.	52
(Tubishat et al., 2019)	A version of IWOA was proposed with two improvements: 1) Elite Opposition-Based Learning (EOBL) at WOA initialisation phase and 2) Differential Evolution (DE) (Storn & Price, 1997) that involved evolutionary operators, i.e., mutation, crossover, and selection at the end of each iteration. The proposed method removes irrelevant features by using IG (Song et al., 2011) in the first phase and then applies IWOA. The SVM classification accuracy The maximum number of iterations (40) Superiority of this method over the GA, PSO, GWO, GSA, and MFO showed that HGSA on 18 well-known UCI datasets.	1) DE evolutionary operators enhance the local search capability of WOA. 2) The initialisation phase of WOA is improved by EBOL.	53

(*Continued*)

Table 16. (Continued).

Publication	Methodology Fitness function Stopping Criteria Experimental results	Findings	Key NO.
(Nematzadeh et al., 2019)	The Whale Mutual Congestion (WMC) method has three phase. In the first phase, WOA is used to remove 50% of the irrelevant and less relevant features. In the second phase, MC is employed to prioritise and sort the remaining features. MC calculates the frequency of true and false labels in three steps (sorting, identifying samples with true and false labels, and Finding Interference of true and false labels). In the third phase, the majority voting function is performed using forward feature selection with threshold 10 on the best feature subsets obtained in the second phase. $\sqrt{(x-\text{MeanY})^2-(y-\text{MeanX})^2}$, where MeanX and MeanY are mean value of first class and second class, respectively. The maximum number of iterations (100) Superiority of this method over the standard WOA, Mutual Congestion, HS, FA and PSO on four UCI medical datasets.	1) The proposed fitness function of WOA increases the efficiency of the algorithm to remove irrelevant features. 2) MC identifies properly the interference areas of true and false labels and then selects the best features by ranking features based on this interference.	54

Table 17. Details of methodology and findings of the FA-based feature selection algorithms.

Publication	Methodology / Fitness function / Stopping Criteria / Experimental results	Findings	Key NO.
(Sawhney et al., 2018)	Random Forest with Binary FA (called BFA+Penalty+RF) uses a penalty-based fitness function and RF classifier to reach a small-size and optimal subset. In this method, each firefly is represented by a binary vector similar to the Key 3.	The penalty-based fitness function effectively reduces size of selected subset and the computational time in the high-dimensional problems without losing the accuracy.	56
	$Accuracy - \alpha\lvert N\rvert/\lvert S\rvert$, where $\lvert N\rvert$ and $\lvert S\rvert$ are the number of original and selected features, respectively. The weight α is varied from 0.05 to 0.5 and its optimal value is fixed on 0.12.		
	The maximum number of iterations (200)		
	Superiority of this method over the BFFA and BPSO on three UCI medical datasets.		
(L. Zhang et al., 2018)	The prop.FA employs four strategies: 1) the Logistic chaotic map movements at the first phase, 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 towards strong-light fireflies and the weak solutions towards optimal regions.	This method overcomes premature convergence and reach the global optima by employing mentioned strategies.	57
	$\alpha Accuracy - (1-\alpha)(\lvert S\rvert)^{-1}$, $\lvert S\rvert$ is the size of subset, and $\alpha = 0.9$.		
	The maximum number of iterations or finding the optimal solutions		
	Superiority of this method over the SA, BSO, CS, PSO, DA, ALO, GA, TS, Memetic Algorithm with Local Search chains (MA-LS), and Differential Evolution (DE) on 29 classification and 11 regression datasets.		
(Selvakumar & Muneeswaran, 2019)	A filter and wrapper-based FA feature selection method was proposed for Intrusion Detection System (IDS). In this method, three different methods are used: MI, Mutual Information-based Firefly Algorithm (MIFA) with C4.5 (Heckerman, 2008), MIFA wrapper method with Bayesian network (Quinlan, 2014). Then, a Voting-based feature selection is used to select final features subset.	The ten selected features using this method are sufficient to detect intrusion and can improve accuracy.	58
	The classification accuracy		
	The maximum number of iterations (100)		
	Superiority of this method over the MI, MIFA with C4.5, or MIFA with the Bayesian network on the KDD CUP 99 dataset		
(Alfarraj et al., 2019)	The two-steps FGACO-based feature selection method uses Min-Max along with z-score normalisation (Z. Zhang et al., 2014) in the first step to eliminate the noisy data. Then, the Firefly Gravitational ACO (FGACO) method is employed to find the optimal subset. The FA attractiveness and intensity values are calculated by the Minkowski distance and the minimum and maximum values of the feature. Additionally, the acceleration, velocity, and mass of the recognised feature are computed based on Newton's law.	1) Low runtime and cost 2) Solving the convergence and local optima problems.	59
	$F_{ij}^{d}(t) = G\_force(t)\left(Mass_i(t)M_{assj}(t)\right)/\left(\left(Dis_{ij}(t)\right)^{n} + \varepsilon\right)$		
	Finding the optimal solutions		
	Superiority of this method over the GSA, FA, and ACO on four UCI datasets.		

Table 18. Details of methodology and findings of the FFOA-based feature selection algorithms.

Publication	Methodology / Fitness function / Stopping Criteria / Experimental results	Findings	Key NO.
(Hou et al., 2019)	The Binary Improved FFOA (BIFFOA) was proposed based on Improved FFOA with dynamic search radius (Q. -K. Pan et al., 2014), the Evolutionary Population Dynamics (EPD) (M. Mafarja et al., 2018), a selection operator and a Sigmoid transfer function. In BIFFOA_EPD, the poorer flies are placed around the best three solutions and a randomly-generated solution with the same probability. BIFFOA_EPD_CM uses the mutation operator to mutate the selected solution and the crossover operator (Faris et al., 2018) to crossover the mutated solution with the weak solution. In BIFFOA_EPD_Tour, the best solution is selected by using TS operator. Then, the same crossover and mutation operators of BIFFOA_EPD_CM is performed to the obtained solution. BIFFOA_EPD_RWS uses the Roulette Wheel Selection (RWS) instead of the TS operator in BIFFOA_EPD_Tour and the solutions are selected based on their fitness values. Like Key 13 The maximum number of iterations (100) Superiority of this method over the GA, PSO, bGWO, CFS, IG, Spectrum, and F-Score on 25 UCI datasets.	1) BIFFOA is suitable for high-dimensional datasets. 2)BIFFOA_EPD_Tour is better than the other proposed EPD mechanisms in feature selection problems. 3) Use of the TS operator in BIFFOA_EPD_Tour enhances the population diversity.	60
(X. Zhang et al., 2020)	A new version of FOA (MCFOA) was proposed which uses the Gaussian mutation operator to increase the diversity of FOA and avoid premature convergence, and the chaotic local search strategy (CFOA) to improve exploitation of potential regions. Like Key 13 The maximum number of iterations (1000) Superiority of this method over the BGWO, BA, and BPSO on 7 distinct UCI datasets.	1) A coordinated balancing between the global search and local disturbance by using the Gaussian mutation operator. 2) Reaching to the better position by use of Chaotic local search.	61

existing individuals, disrupting the cluster centres, and creating solutions. Table 19 summarises the primary details taken from recent articles published on the BSO-based feature selection.

The stability and robustness of FAM-BSO in high dimensional datasets is low.

cuttlefish Algorithm-based feature selection

Cuttlefish Algorithm (CFA) was proposed by (Eesa et al., 2014) that mimics the colour changing mechanism of the cuttlefish. This algorithm considers two processes: Reflection (simulation of the light reflection mechanism) and visibility (simulation of the similar pattern), which serve as a strategy for searching the new solutions. Table 20 presents the primary details taken from studies recently conducted on CFA-based feature selection.

The performance and stability of FGLCC-CFA have not been studied in other high dimensional datasets.

Table 19. Details of methodology and findings of the BSO-based feature selection algorithms.

Publication	Methodology / Fitness function / Stopping Criteria / Experimental results	Findings	Key NO.
(Pourpanah et al., 2019)	The hybrid FAM-BSO combines the Fuzzy ARTMAP (FAM) (Carpenter et al., 1992) with Brain storm optimisation (BSO). The FAM-BSO has two phases: learning phase that uses FAM as an incremental learning neural network for training, and feature selection phase that uses BSO to select the best feature subset with the aim of increasing the accuracy and reducing the complexity. The classification error Not mentioned Superiority of this method over the basic FAM, PSO, ACO, GA, GP, FAM-GA, and FAM-PSO on 10 UCI datasets and a human motion detection case study.	FAM-BSO uses the concept of 'open prototype' to select the best features.	62

Emperor Penguin Optimizer-based feature selection

The Emperor Penguin Optimizer (EPO) first presented by (Dhiman & Kumar, 2018) is a meta-heuristic algorithm that mimics the emperor penguins' huddling behaviours to solve the engineering optimisation problems. EPO has four main steps: generating the huddle boundary, computing the temperature around the huddle, calculating the distance, and finding the efficient mover. Table 21 demonstrates the primary details taken from the recent papers on the EPO-based feature selection.

BEPO is not suitable for very large datasets.

Table 20. Details of methodology and findings of the CFA-based feature selection algorithms.

Publication	Methodology / Fitness function / Stopping Criteria / Experimental results	Findings	Key NO.
(Mohammadi et al., 2019)	Feature Grouping based on the Linear Correlation Coefficient- Cuttlefish Algorithm (FGLCC-CFA) is an IDS-based feature selection that combines Feature Grouping based on the Linear Correlation Coefficient (FGLCC) (Ambusaidi et al., 2016) as a filter method and CFA as a wrapper method. The FGLCC-CFA uses the high speed of FGLCC and the high accuracy of CFA to select the best feature subset. $\alpha \cdot DR + (1-\alpha)\cdot(100 - FPR)$, where $DR = TP/(TP+FN)$, $FPR = FP/(FP+TN)$, and $\alpha = 0.7$. Finding the optimal solutions Superiority of this method over the Feature Grouping based on MI (FGMI), Linear Correlation coefficient Feature Selection (LCFS), Novel hybrid KPCA and SVM with GA (N-KPCA-GA-SVM), FGLCC and CFA on KDD Cup 99 large datasets with ID3 DT.	The use of FGLCC-CFA significantly reduces false positive rates and increases attack detection and accuracy rates for different types of attacks.	63

Table 21. Details of methodology and findings of the EPO-based feature selection algorithms.

Publication	Methodology / Fitness function / Stopping Criteria / Experimental results	Findings	Key NO
(Dhiman & Kumar, 2018) and	S-shaped Binary EPO and V-shaped Binary EPO was proposed by using four V-shaped functions $\frac{v}{\sqrt{1+v^2}}$, $\lvert \tanh(v) \rvert$, $\left\lvert \frac{2}{\pi} arctan\left(\frac{\pi}{2}v\right)\right\rvert$ and $\mathrm{erf}\left(\frac{\sqrt{\pi}}{2}v\right)$ and four S-shaped functions $1/(1+e^{-2v})$, $1/(1+e^{-v})$, $1/\left(1+e^{(-v/2)}\right)$ and $1/\left(1+e^{(-v/3)}\right)$. Like Key 13 The maximum number of iterations (500) Superiority of BEPO over the BSHO, BWOA, BDA, BBA, BGWO, and BGSA on 12 datasets.	BEPO with transfer function $\lvert (2/\pi) arctan((\pi/2)v) \rvert$ achieved the highest performance compared to the other versions.	64

Analysis and discussion

This section discusses the data set, evaluation measures, classifiers used by the various methods and time complexity mentioned in the previous section.

Dataset

The datasets used by these methods can be divided into three groups based on the dimension of dataset: Small (up to 150), Medium (between 151 and 1999), and Large (2000 and more). Table 22 shows a comparison between the datasets used in these methods in terms of size, field, and the number of classes.

Table 22. Comparison between the datasets.

Comparison criteria	Category	Method
Size	Small	3, 8, 18, 25, 28, 29, 32, 33, 36, 40, 45, 49, 51, 55, 56, 58, 61, 62, 63
	Medium	12, 20, 30
	Large	2, 5, 6, 11, 21, 22, 41, 48, 49, 53, 54, 59
	Mixed	4, 7, 9, 10, 13, 14, 15, 16, 17, 19, 23, 24, 26, 27, 31, 34, 35, 37, 38, 39, 42, 43, 44, 46, 47, 50, 52, 57, 60, 64
Field	Text	11, 26
	Image	10, 12, 26, 39,
	Biological	1, 2, 6, 9, 10, 13, 14, 15, 16, 17, 18, 19, 21, 23, 24, 25, 26, 27, 28, 29, 31, 33, 34, 35, 36, 37, 38, 39, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 52, 54, 55, 56, 57, 59, 60, 61, 64
	Life	1, 3, 4, 5, 7, 9, 10, 13, 14, 15, 16, 17, 19, 20, 23, 24, 25, 28, 31, 32, 33, 34, 35, 36, 37, 38, 42, 43, 44, 45, 46, 47, 49, 50, 51, 52, 55, 57, 59, 60, 61, 62, 64
	Physical	1, 4, 9, 10, 13, 14, 15, 16, 17, 20, 23, 24, 25, 28, 31, 33, 34, 35, 36, 37, 38, 42, 43, 44, 45, 46, 47, 49, 52, 55, 57, 60, 61, 62, 64
	Computer	1, 7, 9, 14, 15, 16, 17, 20, 22, 23, 24, 28, 30, 32, 34, 36, 38, 44, 46, 47, 53, 57, 58, 60, 63, 64
	other	4, 6, 9, 10, 13, 14, 15, 16, 17, 20, 23, 24, 25, 28, 31, 33, 34, 35, 36, 37, 38, 40, 42, 43, 44, 45, 46, 47, 49, 50, 52, 55, 57, 59, 60, 61, 62, 64
NO. of classes	Binary class	6, 11, 18, 22, 27, 30, 32, 54, 56
	Multi-class (>2)	3, 5, 8, 9, 10, 40, 51, 53, 58, 63
	Both	1, 2, 4, 7, 12, 13, 14, 15, 16, 17, 19, 20, 21, 23, 24, 25, 26, 28, 29, 31, 33, 34, 35, 36, 37, 38, 39, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 52, 55, 57, 59, 60, 61, 62, 64

As shown in Figure 2, 29% of the methods were applied to small datasets, 5% to medium datasets, 19% to large datasets, and 47% to multiple groups of datasets.

These feature selection meta-heuristic algorithms have used datasets in different fields. Accordingly, seven groups of fields have been considered in this paper: Text, Image, Biological, life, Physical, computer, and other. As shown in Figure 3, two methods were tested on the Text datasets, four methods on the image datasets, 47 methods on the biological datasets, 43 methods on the life datasets, 35 methods on the Physical datasets, 26 methods on the computer datasets, and 38 methods on the other datasets. It is to be noted that some methods have been applied to different fields of datasets. Therefore, they were counted in all considered fields.

On the other hand, the used datasets can also be evaluated in terms of the number of class labels. As shown in Figure 4, 14% of the methods were implemented on two-class labels datasets, 16% on multi-classes dataset, and 70% on both groups.

Classifier

Any feature selection method was evaluated by its classifier. The classifiers used in the previously-mentioned methods are as follows:

- K-Nearest Neighbour (KNN): KNN is one of the Supervised Machine Learning algorithms. To obtain the similarity of two samples, KNN uses the distance criterion of the two samples. KNN is easy and simple to learn. Its training is very fast and it is resistant to noisy training set. On the other hand, it has high computational complexity and memory limitations, and also depends on K.
- Naive Bayes (NB): The NB classifier is an easy, fast, noise resistant and Bayes theorem-based classifier that assumes the existence of a particular feature in a class independent of the

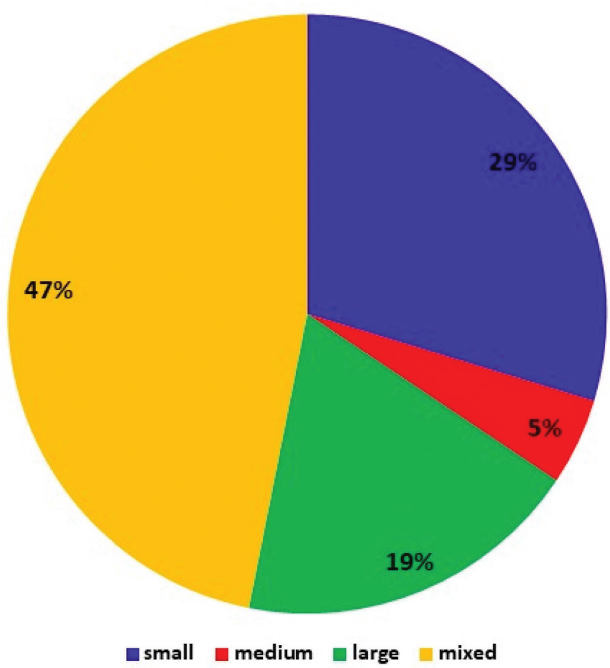

Figure 2. Datasets analysis in terms of no. features.

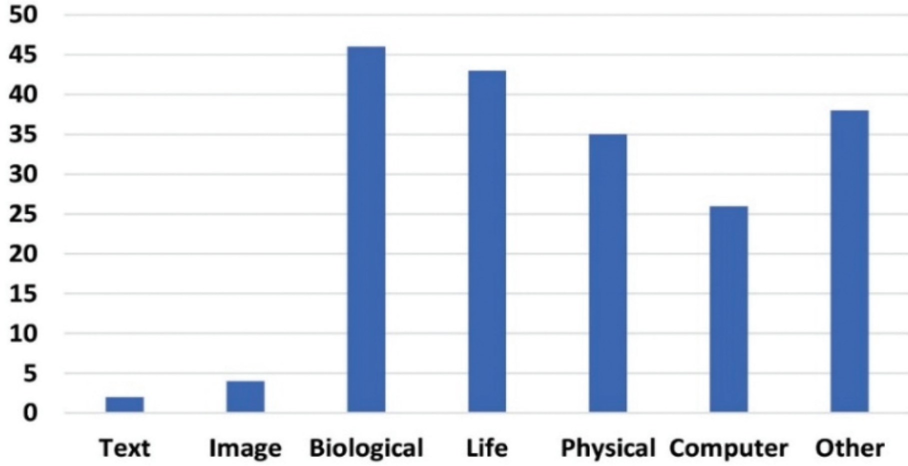

Figure 3. Datasets analysis in terms of fields.

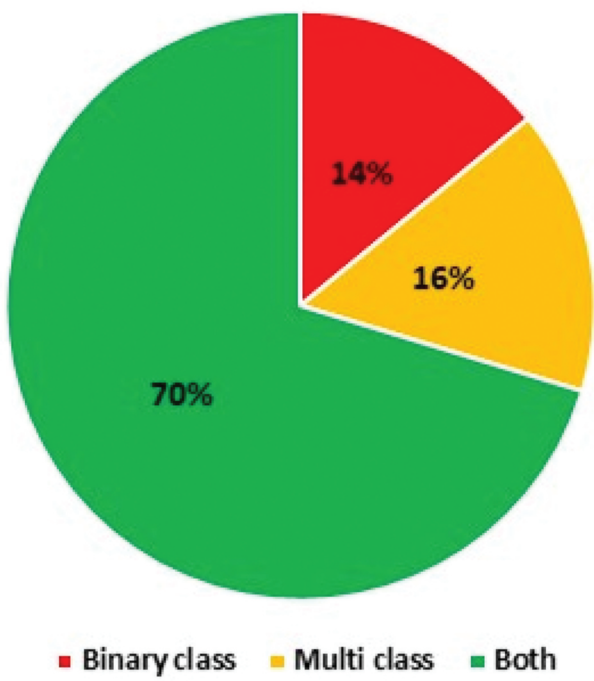

Figure 4. Datasets analysis in terms of NO. class labels.

existence of another feature. The performance of this classifier is high for datasets with more than 2 clusters.

- Support Vector Machine (SVM): SVM is a statistical learning theory-based model that tries to ensure the minimum true error. This classifier is suitable for high-dimensional data and is relatively simple. It also balances between complexity and error rate.

- Decision tree (DT): The decision tree is a predictor model used for regression and classification. This classifier can work on nominal and numeral data. On the other hand, it is suitable for analysing continuous and discrete high-dimensional data in a short amount of time.
- Random Forest (RF): RF is a flexible supervised classifier that is employed in classification and regression problems. This classifier works based on voting between predictions of created decision trees on data samples. The RF usually achieves a better accuracy than a single decision tree, especially when there is a lot of missing data.
- Multi-Layer Perceptron (MLP): MLP (Albu et al., 1997) is one of artificial neural networks structures. Typically, these networks consist of an input layer, one or more hidden layers, and an output layer. The input signal is transmitted layer by layer through the network and in the forward path. After comparing the desired output value with the actual network output, the network searches for the maximum descending slope and in subsequent iterations, the network parameters are adjusted by guiding the descending slope error. The parameters setting is repeated until the amount of network error reaches an acceptable value. Since classification is a specific type of regression, MLP is a desirable classification algorithm when the response variable is definite.
- Other: Extreme Learning Machine (ELM), Linear Discriminant Analysis (LDA), ZeroR, etc.

The classifiers used in the presented methods in this paper are summarised in Table 23.

Figure 5 shows a comparison of the classifiers used in the studies mentioned in this paper. As can be seen, KNN was used in most cases (42 times). SVM is in the second place with 21 cases. DT and NB are in the third and fourth places with 10 and 9 cases, respectively. RF and MLP were used in five studies. And the rest of classifiers are placed in the 'other' category. Note that some methods were analysed by several of these classifiers, which were counted separately.

It should also be noted that 90% of the studies used the k-fold cross-validation method for validation. In this method, the data is converted to k sections. Each observation is randomly placed in one of these k sections. In this method, training and testing are repeated k times. In each iteration $k-1$ part is considered as training data and one as test. The classification error is equal to the average of classification error in k iterations. The advantage of this method is that all observations are used for both training and validation, and each observation is used accurately for validation. This method also has an acceptable speed.

Used methods

The majority of applications, only use a few of the special algorithms. As shown in Table 24, the six most used algorithm are used in more than half of all cases (56.40%), while the eight widely used algorithms comprise three quarters of cases (66.66%), and the top ten algorithms comprise nearly

Table 23. Classifiers used for feature selection algorithms.

Classifier	Methods
SVM	1, 3, 4, 6, 8, 9, 10, 12, 18, 19, 22, 30, 39, 40, 48, 50, 52, 53, 54, 57, 63
KNN	1, 3, 4, 8, 10, 13, 14, 15, 16, 17, 19, 20, 22, 23, 24, 26, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 53, 55, 60, 61, 62
DT	1, 3, 6, 7, 8, 32, 37, 54, 57, 58
NB	1, 6, 8, 21, 22, 39, 53, 54, 58
RF	30, 32, 39, 56, 57
MLP	3, 4, 10, 51, 57
Other	2, 5, 11, 25, 27, 30, 48, 59, 64

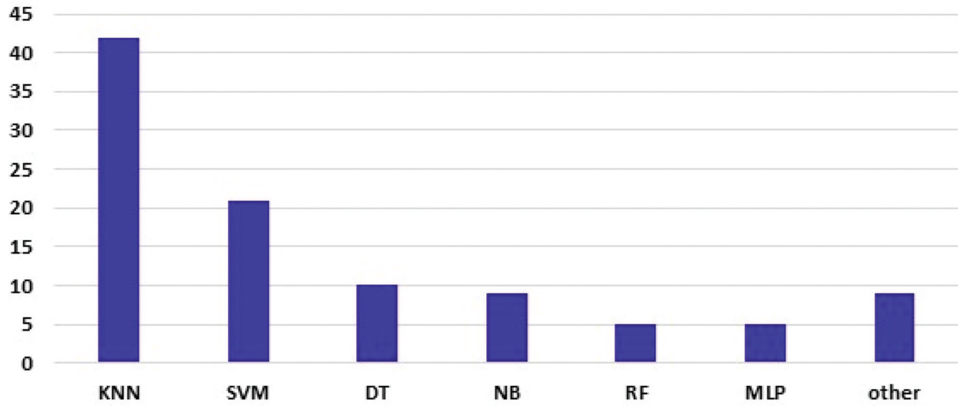

Figure 5. Analysis of the implemented classifiers.

75% of all cases (74.90%). In Table 24, the other category includes eight algorithms: GSA, DA, CRO, SHO, FFOA, BSO and CS.

Table 25 presents the categorisation of 63 selected papers based on different feature selection strategies: Filter, Wrapper, Embedded, and Hybrid. As can be seen, most of the studied methods belong to wrapper methods. Because of training and testing in feature space, these methods perform better than filter-based methods that ignore the effect of selected features on the performance of the machine learning. But Wrapper methods suffer from computational inefficiency. On the other hand, hybrid methods try to reduce the computational load by combining two or more filter or wrapper methods. The same is true for embedded methods. These methods try to use the advantages of both methods by combining filter and wrapper methods and increase the computational speed. however, the existing machine learning algorithm needs to redesigned for embedding feature selection alongside its learning elements, and doing this requires more time and effort.

As shown in Figure 6, 58% of the methods have used the wrapper technique and 30% the hybrid technique. The embedded and filter techniques with 11% and 1%, respectively, are placed at the next levels.

Table 24. Impact (share of citations) of different meta-heuristic algorithms.

Algorithm	Publication	Citations	Shared
SSA	6	1059	13.47
GWO	7	875	11.13
PSO	8	813	10.34
GOA	5	797	10.14
EPO	1	449	5.71
GA	6	443	5.64
CSA	2	432	5.50
FA	4	375	4.77
WOA	4	343	4.36
ACO	4	305	3.88
BOA	2	303	3.86
ABC	1	300	3.82
ALO	3	264	3.36
CFA	1	213	2.71
Other	10	894	11.38

Table 25. The categorisation of methods based on the adopted strategies.

Category	Method
Filter	26
Wrapper	3, 7, 8, 9, 12, 14, 15, 16, 17, 18, 19, 20, 23, 25, 27, 28, 30, 32, 33, 36, 38, 40, 41, 42, 43, 44, 46, 49, 50, 51, 55, 56, 57, 60, 61, 63, 64
Embedded	4, 10, 47, 53, 54, 58, 59
Hybrid	1, 2, 5, 6, 11, 13, 21, 22, 24, 29, 31, 34, 35, 37, 39, 45, 48, 52, 62

Evaluation criteria

Different criteria are used to evaluate the performance and effectiveness of each method in solving the feature selection problem. In the reviewed articles, most of the following evaluation criteria have been used.

• Accuracy: A metric for checking whether samples are correctly classified. The classification accuracy is calculated using Equation(1)

$$Average\_Accuracy = \frac{1}{M}\sum\nolimits_{i=1}^{M}\frac{1}{N}\sum\nolimits_{j=1}^{N} match\left(C_j.L_j\right) \tag{1}$$

where M and N are the number of runs and the number of classes, and *match* is a function of checking the similarity of output class labels and reference class labels.

- Recall or sensitivity: The true positive rate.
- Precision: Positive predictive value (PPV).
- Specificity: True negative rate.
- F-measure: The weighted average between Precision and Recall.

These criteria measures are calculated based on Equations 2–5 as follows:

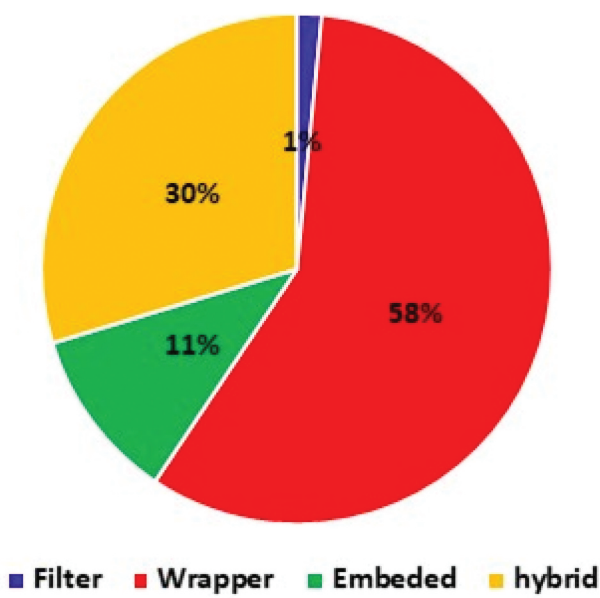

Figure 6. Analysis of the techniques adopted.

$$Specificity = \frac{TN}{TN + FP} \tag{2}$$

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

$$Precision = \frac{TP}{TP + FP} \tag{4}$$

$$F - Measure = 2\frac{Recall*Precision}{(Recall + Precision)} \tag{5}$$

where TP and TN are True positive and True negative, respectively, while FP and FN are false positive and false negative, respectively.

• Average selected subset length: The mean number of selected features in M runs. It is formulated by Equation 6:

$$Average\_Subset\_Length = \frac{1}{M}\sum_{i=1}^{M} Size(S) \tag{6}$$

where S is a feature subset.

• Statistical standard deviation: The variation of the best solutions obtained from M runs. It is formulated by Equation 7:

$$StdDev = \sqrt{\frac{1}{M-1}\sum\left(S^i - STMean\right)^2} \tag{7}$$

where $STMean$ is the mean of solutions in M runs and S^i is the best solution in the ith run.

• Average runtime: The average of actual runtime in milliseconds in M runs. It is calculated by Equation(8:

$$Average\_Runtime = \frac{1}{M}\sum_{i=1}^{M} Runtime_i \tag{8}$$

where $Runtime_i$ is the actual computational time of method at the ith run.

- Wilcoxon test: A non-parametric statistical test based on the mean accuracy for checking statistically difference of two methods. This test returns a number (p-value). $p\_value < 0.05$ indicates significance statistical difference between the two examined methods.
- Friedman test: A non-parametric test based on the mean rankings of test treatments to check the significant difference between two methods. The significance level is set to 0.05.
- Best Fitness, Worst Fitness, and Average Fitness
- Other: Acceleration rate (AR), post-hoc test, Iman – Davenport test, T-test, Stability, etc.

Table 26 illustrates the evaluation measures used in the selected methods. As it can be seen, some methods have used several evaluation criteria. The other category in Table 26 includes Iman – Davenport test, post-hoc test, Acceleration rate (AR), T-test, etc.

Figure 7 compares the different criteria used in the papers to evaluate the proposed methods. Nearly 96% of the papers have used the measurement criteria of accuracy and 94% have used the number of selected features, which are then followed by Runtime, Statistical standard deviation, Fitness analysis, and Wilcoxon test with 56%, 43%, 42%, and 38%. Sensitivity was used in 19% of the papers. Additionally, less than 15% of the papers have used the F-measure, Specificity, and Precision. And 14% of papers have used other evaluation criteria. It should be noted that some methods have used several different criteria for evaluation, which were counted separately.

Table 26. Evaluation measures used in the 64 selected methods.

Evaluation Measure	Methods
Accuracy	All methods except 5, 27 AND 64
Recall	2, 3, 9, 11, 15, 32, 39, 40, 51, 54, 59, 64
Precision	11, 39, 40
Specificity	2, 3, 9, 15, 32, 39, 51, 54, 59
F-measure	11, 14, 27, 39, 40, 45, 58,
No. Selected feature	All methods except 1, 5, 8 and 27
Standard Deviation	9, 10, 14, 15, 16, 17, 19, 25, 28, 31, 33, 34, 35, 36, 37, 38, 39, 40, 42, 43, 47, 49, 52, 55, 61, 62, 64
Average Runtime	1, 6, 7, 9, 10, 12, 13, 14, 15, 17, 21, 24, 27, 28, 29, 30, 31, 35, 36, 37, 39, 40, 43, 44, 45, 46, 47, 48, 49, 51, 54, 55, 58, 59, 60, 63
Wilcoxon Test	12, 15, 16, 17, 24, 26, 28, 30, 33, 34, 35, 36, 37, 38, 39, 41, 42, 45, 47, 50, 52, 57, 61, 64
Friedman Test	2, 7, 13, 17, 23, 42, 34, 38, 61
Best, Worst and Average Fitness	12, 14, 15, 16, 28, 29, 30, 31, 33, 34, 36, 37, 38, 40, 41, 42, 44, 46, 47, 49, 53, 55, 60, 61, 63, 64
Other	5, 14, 23, 25, 27, 39, 45, 59, 63

Time complexity

Although some authors consider the time complexity as a good measure to evaluate and compare their method with other methods (as evident in Table 27), but because meta-heuristic algorithms work based on optimisation using iteration, it is not possible to obtain the time complexity (Note that most of the papers in this field have used meta-heuristics as a part of the total feature selection method, not the whole method). Also, the time in evolutionary algorithms depends on the number of population, the complexity of the fitness function and the number of variables of the problem, which is different in each problem. For example, in paper No. 6, the genetic algorithm with two different fitness functions based on KNN and ANN was used, and KNN achieved a reasonable time compared to ANN. In this way, it may be possible to calculate and compare different time complexities for different meta-heuristics that are defined on a feature selection problem, but it is not logical for different meta-heuristics with different settings and different fitness functions on different feature selection problems.

Conclusions

Nowadays, the interest in using feature selection techniques is increasing because the high-dimensional datasets is a big challenge for patterns recognition, machine learning techniques, data mining and natural language processing. Therefore, a lot of research has been done to choose the best

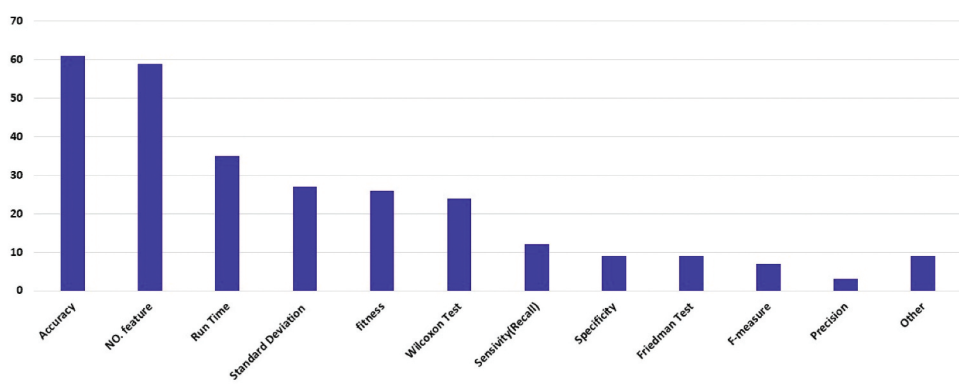

Figure 7. A comparison of the various criteria used in the papers.

Table 27. Time complexity of some studied methods.

Method	Time complexity
1	$O(T \times m \times n \times (m \log n + 1)$
10, 16, 33, 49	$O(T \times m \times n)$
12	$O(m \log m)$
15	$O(T \times m \times n^2 + n/2)$
23	$O(T(nm + n\log_2 n))$
24	$O((T + M_1)(n + M_2)C)$
38, 39, 47	$O(T(m \times n + C \times n))$
48	$O(n^2 + T \times (m^2 + 2n^2))$
52	$O(T^*(m^2 + m^*n + n^*C))$
60	$O(T \times m \times n + n/2)$
61	$O(n \times m + n^2 + T^*(2n^2 + n \times m + 4n))$
62	$O(n^3)$

T: The number of iterations, n: The number of solution or particles, m: The number of features, C: The cost of the objective function, M_1: The logistic map sequence time; M_2: The calculation time for the dynamic correction factors.

features from all available features. The feature selection process improves the speed of learning, simplicity of rules, data visualisation, and predictive accuracy.

This paper presented a comprehensive survey of feature selection meta-heuristic algorithms. Sixty-three different feature selection methods based on 21 meta-heuristic algorithms (proposed between 2018 and 2022) were analysed and described. It was revealed that the majority of meta-heuristic methods seek to maximise the classification accuracy and minimise the number of selected features. They follow these goals by examining the fitness of the obtained solutions. Also, it was found that the majority of these method is wrapper-based method that are computationally expensive when facing a large number of features. Therefore, hybrid feature selection methods were developed that use filter methods to remove redundant and irrelevant features, and then wrapper methods to further modify the selected subset. Although, the meta-heuristic algorithms are the powerful techniques to solve the feature selection problem, there is no clear answer to what is the most effective and perfect meta-heuristic algorithm in feature selection. Because, 1) a general-purpose optimisation strategy is impossible (Wolpert & Macready, 1997), 2) most of the studied methods were lack good empirical evaluations on different types of datasets in different fields and sizes, and 3) it is also very difficult to compare obtained results from different real-world problems. Therefore, the followings are expected for future directions/trends of meta-heuristic algorithms in feature selection:

(1) A comprehensive experimental study on various optimisation algorithms, especially PSO, WOA, BOA, GOA, and DA, which have shown promising feature selection results in high-dimensional datasets, is essential.
(2) Given the emergence of big data with millions of instances and features in the present era, the need to develop more accurate, efficient, and faster feature selection methods is felt. The use of parallel meta-heuristic methods is one of the suggestions that can be proposed.
(3) It is necessary to develop online feature selection methods for real-time feedback.
(4) The quality criteria for an experimental study are numerous, such as accuracy, reliability, stability, feature-class dependency and so on. Therefore, the selection of a unified measurement criteria to evaluate the performance, advantages and disadvantages of studied algorithms seems necessary. In this case, we will certainly have a robust experimental study.

Disclosure statement

No potential conflict of interest was reported by the authors.

ORCID

Ebrahim Akbari http://orcid.org/0000-0002-4721-049X

References

Abasabadi, S., Nematzadeh, H., Motameni, H., & Akbari, E. (2021). Automatic ensemble feature selection using fast non-dominated sorting. *Information Systems*, *100*, 101760. https://doi.org/10.1016/j.is.2021.101760

Abasabadi, S., Nematzadeh, H., Motameni, H., & Akbari, E. (2022). Hybrid feature selection based on SLI and genetic algorithm for microarray datasets. *The Journal of Supercomputing*, *78*(18), 1–29. https://doi.org/10.1007/s11227-022-04650-w

Abdel Aziz, M., & Hassanien, A. E. (2018). Modified cuckoo search algorithm with rough sets for feature selection. *Neural Computing & Applications*, *29*, 925–934. https://doi.org/10.1007/s00521-016-2473-7

Abdel-Basset, M., El-Shahat, D., El-Henawy, I., De Albuquerque, V. H. C., & Mirjalili, S. (2020). A new fusion of grey wolf optimizer algorithm with a two-phase mutation for feature selection. *Expert Systems with Applications*, *139*, 112824. https://doi.org/10.1016/j.eswa.2019.112824

Ain, Q. U., Al-Sahaf, H., Xue, B., & Zhang, M. (2022). Genetic programming for automatic skin cancer image classification. *Expert Systems with Applications*, *197*, 116680. https://doi.org/10.1016/j.eswa.2022.116680

Ain, Q. U., Xue, B., Al-Sahaf, H., & Zhang, M. (2018). Genetic programming for feature selection and feature construction in skin cancer image classification. Pacific rim international conference on artificial intelligence, August 28–31, Nanjing, China, (732–745). Springer. https://doi.org/10.1007/978-3-319-97304-3_56

Al-Helali, B., Chen, Q., Xue, B., & Zhang, M. (2019). Genetic programming for imputation predictor selection and ranking in symbolic regression with high-dimensional incomplete data. Australasian Joint Conference on Artificial Intelligence, Adelaide, SA, Australia, (pp. 523–535). Springer. https://doi.org/10.1007/978-3-030-35288-2_42

Al-Tashi, Q., Kadir, S. J. A., Rais, H. M., Mirjalili, S., & Alhussian, H. (2019). Binary optimization using hybrid grey wolf optimization for feature selection. *IEEE Access*, *7*, 39496–39508. https://doi.org/10.1109/ACCESS.2019.2906757

Albu, F., Mateescu, A., & Dumitriu, N. (1997). Architecture selection for a multilayer feedforward network. International Conference on Microelectronics and Computer Science, Denver, Colorado, (pp. 131–134).

Albuquerque, I. M. R., Nguyen, B. H., Xue, B., & Zhang, M. (2020). A novel genetic algorithm approach to simultaneous feature selection and instance selection. 2020 IEEE Symposium Series on Computational Intelligence (SSCI), Canberra, ACT, Australia, (pp. 616–623). IEEE.

AlfarraJ, O., Alzubi, A., & Tolba, A. (2019). Optimized feature selection algorithm based on fireflies with gravitational ant colony algorithm for big data predictive analytics. *Neural Computing & Applications*, *31*, 1391–1403. https://doi.org/10.1007/s00521-018-3612-0

Aljarah, I., Mafarja, M., Heidari, A. A., Faris, H., Zhang, Y., & Mirjalill, S. (2018). Asynchronous accelerating multi-leader salp chains for feature selection. *Applied Soft Computing*, *71*, 964–979. https://doi.org/10.1016/j.asoc.2018.07.040

Ambusaidi, M. A., He, X., Nanda, P., & Z, T. A. N. (2016). Building an intrusion detection system using a filter-based feature selection algorithm. *IEEE Transactions on Computers*, *65*(10), 2986–2998. https://doi.org/10.1109/TC.2016.2519914

Amini, F., & Hu, G. (2021). A two-layer feature selection method using genetic algorithm and elastic net. *Expert Systems with Applications*, *166*, 114072. https://doi.org/10.1016/j.eswa.2020.114072

Anter, A. M., & M, A. L. I. (2020). Feature selection strategy based on hybrid crow search optimization algorithm integrated with chaos theory and fuzzy c-means algorithm for medical diagnosis problems. *Soft Computing*, *24*(3), 1565–1584. https://doi.org/10.1007/s00500-019-03988-3

Arora, S., & Anand, P. (2019). Binary butterfly optimization approaches for feature selection. *Expert Systems with Applications*, *116*, 147–160. https://doi.org/10.1016/j.eswa.2018.08.051

Arora, S., Singh, H., Sharma, M., Sharma, S., & Anand, P. (2019). A new hybrid algorithm based on grey wolf optimization and crow search algorithm for unconstrained function optimization and feature selection. *IEEE Access*, *7*, 26343–26361. https://doi.org/10.1109/ACCESS.2019.2897325

Arora, S., & Singh, S. (2017). Node localization in wireless sensor networks using butterfly optimization algorithm. *Arabian Journal for Science and Engineering*, *42*, 3325–3335. https://doi.org/10.1007/s13369-017-2471-9

Askarzadeh, A. (2016). A novel metaheuristic method for solving constrained engineering optimization problems: Crow search algorithm. *Computers & Structures*, *169*, 1–12. https://doi.org/10.1016/j.compstruc.2016.03.001

Back, T. (1996). *Evolutionary algorithms in theory and practice: Evolution strategies, evolutionary programming, genetic algorithms*. Oxford university press.

Basu, S., Das, N., Sarkar, R., Kundu, M., Nasipuri, M., & Basu, D. K. (2012). Handwritten Bangla alphabet recognition using an MLP based classifier. 2nd National Conference on Computer Processing, Bangla, Bangladesh, (pp. 285–291).

Bezdek, J. C., Ehrlich, R., & Full, W. (1984). FCM: The fuzzy c-means clustering algorithm. *Computers & Geosciences*, *10*, 191–203. https://doi.org/10.1016/0098-3004(84)90020-7

Bhandari, A. K., Singh, V. K., Kumar, A., & Singh, G. K. (2014). Cuckoo search algorithm and wind driven optimization based study of satellite image segmentation for multilevel thresholding using Kapur's entropy. *Expert Systems with Applications*, *41*, 3538–3560. https://doi.org/10.1016/j.eswa.2013.10.059

Bi, Y., Xue, B., & Zhang, M. (2020a). Evolving deep forest with automatic feature extraction for image classification using genetic programming. International Conference on Parallel Problem Solving from Nature, Leiden, The Netherlands, (pp. 3–18). Springer.

Bi, Y., Xue, B., & Zhang, M. (2020). Genetic programming with a new representation to automatically learn features and evolve ensembles for image classification. *IEEE Transactions on Cybernetics*, *51*(4), 1769–1783. https://doi.org/10.1109/TCYB.2020.2964566

Bi, Y., Xue, B., & Zhang, M. (2021). Multi-objective genetic programming for feature learning in face recognition. *Applied Soft Computing*, *103*, 107152. https://doi.org/10.1016/j.asoc.2021.107152

Bi, Y., Zhang, M., & B, X. U. E. (2018). Genetic programming for automatic global and local feature extraction to image classification. 2018 IEEE Congress on Evolutionary Computation (CEC), Rio de Janeiro, Brazil, (pp. 1–8). IEEE.

Brezočnik, L., Fister, I., & Podgorelec, V. (2018). Swarm intelligence algorithms for feature selection: A review. *Applied Sciences*, *8*, 1521. https://doi.org/10.3390/app8091521

Brown, E. C., & Sumichrast, R. T. (2005). Evaluating performance advantages of grouping genetic algorithms. *Engineering Applications of Artificial Intelligence*, *18*, 1–12. https://doi.org/10.1016/j.engappai.2004.08.024

Bryll, R., Gutierrez-Osuna, R., & Quek, F. (2003). Attribute bagging: Improving accuracy of classifier ensembles by using random feature subsets. *Pattern Recognition*, *36*, 1291–1302. https://doi.org/10.1016/S0031-3203(02)00121-8

Carpenter, G. A., Grossberg, S., Markuzon, N., Reynolds, J. H., & Rosen, D. B. (1992). Fuzzy ARTMAP: A neural network architecture for incremental supervised learning of analog multidimensional maps. *IEEE Transactions on Neural Networks*, *3*, 698–713. https://doi.org/10.1109/72.159059

Chechkin, A., Metzler, R., Klafter, J., Gonchar, V. Y., & Sokolov, I. M. (2008). Generalized fractional diffusion equations for accelerating subdiffusion and truncated Lévy flights. *Physical Review E, Statistical, Nonlinear, and Soft Matter Physics*, *78*(2 Pt 1). https://doi.org/10.1103/PhysRevE.78.021111

Chen, K., Zhou, F. -Y., & Yuan, X. -F. (2019). Hybrid particle swarm optimization with spiral-shaped mechanism for feature selection. *Expert Systems with Applications*, *128*, 140–156. https://doi.org/10.1016/j.eswa.2019.03.039

Chowdhury, A. A., Borkar, V. S., & Birajdar, G. K. (2020). Indian language identification using time-frequency image textural descriptors and GWO-based feature selection. *Journal of Experimental & Theoretical Artificial Intelligence*, *32*, 111–132. https://doi.org/10.1080/0952813X.2019.1631392

Dash, M., & H, L. I. U. (1997). Feature selection for classification. *Intelligent Data Analysis*, *1*(3), 131–156. https://doi.org/10.3233/IDA-1997-1302

Deng, X., Li, Y., Weng, J., & Zhang, J. (2019). Feature selection for text classification: A review. *Multimedia Tools & Applications*, *78*(3), 3797–3816. https://doi.org/10.1007/s11042-018-6083-5

Dhiman, G., & Kumar, V. (2017). Spotted hyena optimizer: A novel bio-inspired based metaheuristic technique for engineering applications. *Advances in Engineering Software*, *114*, 48–70. https://doi.org/10.1016/j.advengsoft.2017.05.014

Dhiman, G., & Kumar, V. (2018). Emperor penguin optimizer: A bio-inspired algorithm for engineering problems. *Knowledge-Based Systems*, *159*, 20–50. https://doi.org/10.1016/j.knosys.2018.06.001

Dong, H., Li, T., Ding, R., & J, S. U. N. (2018). A novel hybrid genetic algorithm with granular information for feature selection and optimization. *Applied Soft Computing*, *65*, 33–46. https://doi.org/10.1016/j.asoc.2017.12.048

Dorigo, M., & Gambardella, L. M. (1997). Ant colony system: A cooperative learning approach to the traveling salesman problem. *IEEE Transactions on Evolutionary Computation*, *1*, 53–66. https://doi.org/10.1109/4235.585892

Dueck, G. (1993). New optimization heuristics: The great deluge algorithm and the record-to-record travel. *Journal of Computational Physics*, *104*, 86–92. https://doi.org/10.1006/jcph.1993.1010

Eberhart, R., & Kennedy, J. (1995). Particle swarm optimization. Proceedings of the IEEE international conference on neural networks, Perth, WA, Australia, (pp. 1942–1948). Citeseer.

Eesa, A. S., Brifcani, A. M. A., & Orman, Z. (2014). A new tool for global optimization problems-cuttlefish algorithm. *International Journal of Mathematical, Computational, Natural and Physical Engineering*, *8*(9), 1208–1211.

Emary, E., & Zawbaa, H. M. (2019). Feature selection via Lèvy Antlion optimization. *Pattern Analysis and Applications*, *22*, 857–876. https://doi.org/10.1007/s10044-018-0695-2

Emary, E., Zawbaa, H. M., & Hassanien, A. E. (2016). Binary grey wolf optimization approaches for feature selection. *Neurocomputing*, *172*, 371–381. https://doi.org/10.1016/j.neucom.2015.06.083

Engelbrecht, A. P., Grobler, J., & Langeveld, J. (2019). Set based particle swarm optimization for the feature selection problem. *Engineering Applications of Artificial Intelligence*, *85*, 324–336. https://doi.org/10.1016/j.engappai.2019.06.008

Eusuff, M., Lansey, K., & Pasha, F. (2006). Shuffled frog-leaping algorithm: A memetic meta-heuristic for discrete optimization. *Engineering Optimization*, *38*, 129–154. https://doi.org/10.1080/03052150500384759

Fan, Q., Bi, Y., Xue, B., & Zhang, M. (2022). Genetic programming for feature extraction and construction in image classification. *Applied Soft Computing*, *118*, 108509. https://doi.org/10.1016/j.asoc.2022.108509

Faris, H., Mafarja, M. M., Heidari, A. A., Aljarah, I., Ala'm, A. -Z., Mirjalili, S., & Fujita, H. (2018). An efficient binary salp swarm algorithm with crossover scheme for feature selection problems. *Knowledge-Based Systems*, *154*, 43–67. https://doi.org/10.1016/j.knosys.2018.05.009

Fukushima, A., Sugimoto, M., Hiwa, S., & Hiroyasu, T. (2019). Elastic net-based prediction of IFN-β treatment response of patients with multiple sclerosis using time series microarray gene expression profiles. *Scientific Reports*, *9*, 1–11. https://doi.org/10.1038/s41598-018-38441-2

Gao, W., Hu, L., Zhang, P., & Wang, F. (2018). Feature selection by integrating two groups of feature evaluation criteria. *Expert Systems with Applications*, *110*, 11–19. https://doi.org/10.1016/j.eswa.2018.05.029

Ghimatgar, H., Kazemi, K., Helfroush, M. S., & Aarabi, A. (2018). An improved feature selection algorithm based on graph clustering and ant colony optimization. *Knowledge-Based Systems*, *159*, 270–285. https://doi.org/10.1016/j.knosys.2018.06.025

Ghosh, M., Guha, R., Sarkar, R., & Abraham, A. (2019). A wrapper-filter feature selection technique based on ant colony optimization. *Neural Computing & Applications*, *32*(12), 1–19. https://doi.org/10.1007/s00521-019-04171-3

Goldberg, D. E., & Holland, J. H. (1988). Genetic algorithms and machine learning.

Gu, S., Cheng, R., & Y, J. I. N. (2018). Feature selection for high-dimensional classification using a competitive swarm optimizer. *Soft Computing*, *22*(3), 811–822. https://doi.org/10.1007/s00500-016-2385-6

Guha, R., Ghosh, M., Kapri, S., Shaw, S., Mutsuddi, S., Bhateja, V., & Sarkar, R. (2019). Deluge based genetic algorithm for feature selection. *Evolutionary Intelligence*, *14* (2), 1–11. *Evolutionary intelligence*. https://doi.org/10.1007/s12065-019-00218-5

Hamidzadeh, J. (2021). Feature selection by using chaotic cuckoo optimization algorithm with levy flight, opposition-based learning and disruption operator. *Soft Computing*, *25*(4), 2911–2933. https://doi.org/10.1007/s00500-020-05349-x

Hammami, M., Bechikh, S., Hung, C. -C., & Said, L. B. (2020a). Class-dependent weighted feature selection as a bi-level optimization problem. International Conference on Neural Information Processing, Bangkok, Thailand, (pp. 269–278). Springer.

Hammami, M., Bechikh, S., Hung, C. -C., & Said, L. B. (2018). A multi-objective hybrid filter-wrapper evolutionary approach for feature construction on high-dimensional data. 2018 IEEE Congress on Evolutionary Computation (CEC), Rio de Janeiro, Brazil, (pp. 1–8). IEEE.

Hammami, M., Bechikh, S., Hung, C. -C., & Said, L. B. (2019). A multi-objective hybrid filter-wrapper evolutionary approach for feature selection. *Memetic Computing*, *11*, 193–208. https://doi.org/10.1007/s12293-018-0269-2

Hammami, M., Bechikh, S., Louati, A., Makhlouf, M., & Said, L. B. (2020). Feature construction as a bi-level optimization problem. *Neural Computing & Applications*, *32*, 13783–13804. https://doi.org/10.1007/s00521-020-04784-z

Hammami, M., Bechikh, S., Makhlouf, M., Hung, C. -C., & Said, L. B. (2020c). Class dependent feature construction as a bi-level optimization problem. 2020 IEEE Congress on Evolutionary Computation (CEC), Glasgow, UK, (pp. 1–8). IEEE.

Hancer, E., Xue, B., Karaboga, D., & Zhang, M. (2015). A binary ABC algorithm based on advanced similarity scheme for feature selection. *Applied Soft Computing*, *36*, 334–348. https://doi.org/10.1016/j.asoc.2015.07.023

Hancer, E., Xue, B., & Zhang, M. (2020). A survey on feature selection approaches for clustering. *Artificial Intelligence Review*, *53*, 4519–4545. https://doi.org/10.1007/s10462-019-09800-w

Hancer, E., Xue, B., & Zhang, M. (2022). Fuzzy filter cost-sensitive feature selection with differential evolution. *Knowledge-Based Systems*, *241*, 108259. https://doi.org/10.1016/j.knosys.2022.108259

Heckerman, D. (2008). A Tutorial on Learning with Bayesian Networks. In M. I. Jordan (Ed.), *Learning in Graphical Models* (pp. 301–354). Springer. https://doi.org/10.1007/978-94-011-5014-9_11

Hegazy, A. E., Makhlouf, M., & El-Tawel, G. S. (2019). Feature selection using chaotic salp swarm algorithm for data classification. *Arabian Journal for Science and Engineering*, *44*, 3801–3816. https://doi.org/10.1007/s13369-018-3680-6

Hegazy, A. E., Makhlouf, M., & El-Tawel, G. S. (2020). Improved salp swarm algorithm for feature selection. *Journal of King Saud University-Computer and Information Sciences*, *32*(3), 335–344. https://doi.org/10.1016/j.jksuci.2018.06.003

Hichem, H., Elkamel, M., Rafik, M., Mesaaoud, M. T., & Ouahiba, C. (2019). A new binary grasshopper optimization algorithm for feature selection problem. *Journal of King Saud University-Computer and Information Sciences*, *34*(2), 316–328.

Holland, J. H. (1973). Genetic algorithms and the optimal allocation of trials. *SIAM Journal on Computing*, *2*, 88–105. https://doi.org/10.1137/0202009

Hou, Y., Li, J., Yu, H., & Li, Z. (2019). BIFFOA: A novel binary improved fruit fly algorithm for feature selection. *IEEE Access*, *7*, 81177–81194. https://doi.org/10.1109/ACCESS.2019.2917502

Ibrahim, H. T., Mazher, W. J., Ucan, O. N., & Bayat, O. (2019). A grasshopper optimizer approach for feature selection and optimizing SVM parameters utilizing real biomedical data sets. *Neural Computing & Applications*, *31*, 5965–5974. https://doi.org/10.1007/s00521-018-3414-4

Ibrahim, R. A., Ewees, A. A., Oliva, D., Abd Elaziz, M., & Lu, S. (2019). Improved salp swarm algorithm based on particle swarm optimization for feature selection. *Journal of Ambient Intelligence and Humanized Computing*, *10*, 3155–3169. https://doi.org/10.1007/s12652-018-1031-9

Jain, I., Jain, V. K., & Jain, R. (2018). Correlation feature selection based improved-binary particle swarm optimization for gene selection and cancer classification. *Applied Soft Computing*, *62*, 203–215. https://doi.org/10.1016/j.asoc.2017.09.038

Jayaraman, V., & Sultana, H. P. (2019). Artificial gravitational cuckoo search algorithm along with particle bee optimized associative memory neural network for feature selection in heart disease classification. *Journal of Ambient Intelligence and Humanized Computing*, 1–10. https://doi.org/10.1007/s12652-019-01193-6

Jensen, R., & Shen, Q. (2003). Finding rough set reducts with ant colony optimization. Proceedings of the 2003 UK workshop on computational intelligence, United Kingdom, (pp. 15–22).

Jia, H., Li, J., Song, W., Peng, X., Lang, C., & Li, Y. (2019). Spotted hyena optimization algorithm with simulated annealing for feature selection. *IEEE Access*, *7*, 71943–71962. https://doi.org/10.1109/ACCESS.2019.2919991

Karaboga, D. (2005). *An idea based on honey bee swarm for numerical optimization*. Technical report-tr06. Erciyes university, engineering faculty, computer

Kashani, S. M. Z., & Hamidzadeh, J. (2020). Feature selection by using privacy-preserving of recommendation systems based on collaborative filtering and mutual trust in social networks. *Soft Computing*, *24*, 11425–11440. https://doi.org/10.1007/s00500-019-04605-z.

Kaufman, L., & Rousseeuw, P. J. (2009). *Finding groups in data: An introduction to cluster analysis*. John Wiley & Sons.

Kennedy, J., & Eberhart, R. C. (1997). A discrete binary version of the particle swarm algorithm. 1997 IEEE International conference on systems, man, and cybernetics. Computational cybernetics and simulation, Orlando, FL, USA, (pp. 4104–4108). IEEE.

Khan, A., & Baig, A. R. (2016). Multi-objective feature subset selection using mRMR based enhanced ant colony optimization algorithm (mRMR-EACO). *Journal of Experimental & Theoretical Artificial Intelligence*, *28*, 1061–1073. https://doi.org/10.1080/0952813X.2015.1056240

Kohavi, R., & John, G. H. (1997). Wrappers for feature subset selection. *Artificial Intelligence*, *97*, 273–324. https://doi.org/10.1016/S0004-3702(97)00043-X

Kothari, V., Anuradha, J., Shah, S., & Mittal, P. (2011). A survey on particle swarm optimization in feature selection. International Conference on Computing and Communication Systems, Vellore, TN, India, (pp. 192–201). Springer.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, *60*(6), 84–90. https://doi.org/10.1145/3065386

Kumar, L., & Bharti, K. K. (2019). *An improved BPSO algorithm for feature selection. recent trends in communication, computing, and electronics*. Springer.

Langeveld, J., & Engelbrecht, A. P. (2012). Set-based particle swarm optimization applied to the multidimensional knapsack problem. *Swarm Intelligence*, *6*, 297–342. https://doi.org/10.1007/s11721-012-0073-4

Lewis, A., Mostaghim, S., & Randall, M. (2008). *Evolutionary population dynamics and multi-objective optimisation problems. Multi-objective optimization in computational intelligence: Theory and practice*. IGI Global.

Li, A. -D., Xue, B., & Zhang, M. (2021). Improved binary particle swarm optimization for feature selection with new initialization and search space reduction strategies. *Applied Soft Computing*, *106*, 107302. https://doi.org/10.1016/j.asoc.2021.107302

Li, J., Cheng, K., Wang, S., Morstatter, F., Trevino, R. P., Tang, J., & H, L. I. U. (2017). Feature selection: A data perspective. *ACM Computing Surveys (CSUR)*, *50*(6), 1–45. https://doi.org/10.1145/3136625

Lipowski, A., & Lipowska, D. (2012). Roulette-wheel selection via stochastic acceptance. *Physica A: Statistical Mechanics and Its Applications*, *391*, 2193–2196. https://doi.org/10.1016/j.physa.2011.12.004

Liu, S., Tai, H., Ding, Q., Li, D., Xu, L., & Y, W. E. I. (2013). A hybrid approach of support vector regression with genetic algorithm optimization for aquaculture water quality prediction. *Mathematical and Computer Modelling*, *58*(3–4), 458–465. https://doi.org/10.1016/j.mcm.2011.11.021

Liu, X. -Y., Liang, Y., Wang, S., Yang, Z. -Y., & Ye, H. -S. (2018). A hybrid genetic algorithm with wrapper-embedded approaches for feature selection. *IEEE Access*, *6*, 22863–22874. https://doi.org/10.1109/ACCESS.2018.2818682

Mafarja, M., Aljarah, I., Faris, H., Hammouri, A. I., Ala'm, A. -Z., & Mirjalili, S. (2019). Binary grasshopper optimisation algorithm approaches for feature selection problems. *Expert Systems with Applications*, *117*, 267–286. https://doi.org/10.1016/j.eswa.2018.09.015

Mafarja, M., Aljarah, I., Heidari, A. A., Hammouri, A. I., Faris, H., Ala'm, A. -Z., & Mirjalili, S. (2018). Evolutionary population dynamics and grasshopper optimization approaches for feature selection problems. *Knowledge-Based Systems*, *145*, 25–45. https://doi.org/10.1016/j.knosys.2017.12.037

Mafarja, M., Heidari, A. A., Faris, H., Mirjalili, S., & Aljarah, I. (2020). Dragonfly algorithm: Theory, literature review, and application in feature selection. In S. Mirjalili, J. Song Dong, & A. Lewis (Eds.), *Nature-inspired optimizers* (pp. 47–67). Springer. https://doi.org/10.1007/978-3-030-12127-3_4

Mafarja, M., Qasem, A., Heidari, A. A., Aljarah, I., Faris, H., & Mirjalili, S. (2020). Efficient hybrid nature-inspired binary optimizers for feature selection. *Cognitive Computation*, *12*, 150–175. https://doi.org/10.1007/s12559-019-09668-6

Mafarja, M. M., & Mirjalili, S. (2019). Hybrid binary ant lion optimizer with rough set and approximate entropy reducts for feature selection. *Soft Computing*, *23*, 6249–6265. https://doi.org/10.1007/s00500-018-3282-y

Manoj, R. J., Praveena, M. A., & Vijayakumar, K. (2019). An ACO–ANN based feature selection algorithm for big data. *Cluster Computing*, *22*(S2), 3953–3960. https://doi.org/10.1007/s10586-018-2550-z

Miller, B. L., & Goldberg, D. E. (1995). Genetic algorithms, tournament selection, and the effects of noise. *Complex Systems*, *9*(3), 193–212.

Mirjalili, S. (2015). The ant lion optimizer. *Advances in Engineering Software*, *83*, 80–98. https://doi.org/10.1016/j.advengsoft.2015.01.010

Mirjalili, S. (2016). Dragonfly algorithm: A new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems. *Neural Computing & Applications*, *27*, 1053–1073. https://doi.org/10.1007/s00521-015-1920-1

Mirjalili, S., Gandomi, A. H., Mirjalili, S. Z., Saremi, S., Faris, H., & Mirjalili, S. M. (2017). Salp swarm algorithm: A bio-inspired optimizer for engineering design problems. *Advances in Engineering Software*, *114*, 163–191. https://doi.org/10.1016/j.advengsoft.2017.07.002

Mirjalili, S., & Lewis, A. (2016). The whale optimization algorithm. *Advances in Engineering Software*, *95*, 51–67. https://doi.org/10.1016/j.advengsoft.2016.01.008

Mirjalili, S., Mirjalili, S. M., & Lewis, A. (2014). Grey wolf optimizer. *Advances in Engineering Software*, *69*, 46–61. https://doi.org/10.1016/j.advengsoft.2013.12.007

Mohammadi, S., Mirvaziri, H., Ghazizadeh-Ahsaee, M., & Karimipour, H. (2019). Cyber intrusion detection by combined feature selection algorithm. *Journal of Information Security and Applications*, *44*, 80–88. https://doi.org/10.1016/j.jisa.2018.11.007

Moradi, P., & Rostami, M. (2015). Integration of graph clustering with ant colony optimization for feature selection. *Knowledge-Based Systems*, *84*, 144–161. https://doi.org/10.1016/j.knosys.2015.04.007

Mullick, S. S., Datta, S., Dhekane, S. G., & S, D. A. S. (2020). Appropriateness of performance indices for imbalanced data classification: An analysis. *Pattern Recognition*, *102*, 107197. https://doi.org/10.1016/j.patcog.2020.107197

Nematzadeh, H., Enayatifar, R., Mahmud, M., & Akbari, E. (2019). Frequency based feature selection method using whale algorithm. *Genomics*, *111*, 1946–1955. https://doi.org/10.1016/j.ygeno.2019.01.006

Nguyen, B. H., Xue, B., & Zhang, M. (2020). A survey on swarm intelligence approaches to feature selection in data mining. *Swarm and Evolutionary Computation*, *54*, 100663. https://doi.org/10.1016/j.swevo.2020.100663

Nguyen, H. B., Xue, B., & Andreae, P. (2017). A hybrid GA-GP method for feature reduction in classification. Asia-Pacific Conference on Simulated Evolution and Learning, Shenzhen, China, (pp. 591–604). Springer.

Omran, M. G., & Mahdavi, M. (2008). Global-best harmony search. *Applied Mathematics and Computation*, *198*, 643–656. https://doi.org/10.1016/j.amc.2007.09.004

Ott, M., Choi, Y., Cardie, C., & Hancock, J. T. (2011). Finding deceptive opinion spam by any stretch of the imagination. Annual Meeting of the Association for Computational Linguistics, Portland, Oregon, USA. Association for Computational Linguistics.

Pan, Q. -K., Sang, H. -Y., Duan, J. -H., & Gao, L. (2014). An improved fruit fly optimization algorithm for continuous function optimization problems. *Knowledge-Based Systems*, *62*, 69–83. https://doi.org/10.1016/j.knosys.2014.02.021

Pan, W. -T. (2012). A new fruit fly optimization algorithm: Taking the financial distress model as an example. *Knowledge-Based Systems*, *26*, 69–74. https://doi.org/10.1016/j.knosys.2011.07.001

Pathak, Y., Arya, K., & Tiwari, S. (2019). Feature selection for image steganalysis using levy flight-based grey wolf optimization. *Multimedia Tools and Applications*, *78*, 1473–1494. https://doi.org/10.1007/s11042-018-6155-6

Pei, W., Xue, B., Shang, L., & Zhang, M. (2018). Genetic programming based on granular computing for classification with high-dimensional data. Australasian Joint Conference on Artificial Intelligence, Wellington, New Zealand, (pp. 643–655). Springer.

Pei, W., Xue, B., Shang, L., & Zhang, M. (2021). Genetic programming for development of cost-sensitive classifiers for binary high-dimensional unbalanced classification. *Applied Soft Computing*, *101*, 106989. https://doi.org/10.1016/j.asoc.2020.106989

Pei, W., Xue, B., Shang, L., & Zhang, M. (2022). High-dimensional unbalanced binary classification by genetic programming with multi-criterion fitness evaluation and selection. *Evolutionary Computation*, *30*, 99–129. https://doi.org/10.1162/evco_a_00304

Peng, B., Wan, S., Bi, Y., Xue, B., & Zhang, M. (2020a). Automatic feature extraction and construction using genetic programming for rotating machinery fault diagnosis. *IEEE Transactions on Cybernetics*, *51*(10), 4909–4923.

Peng, B., Wan, S., Bi, Y., Xue, B., & Zhang, M. (2020b). Automatic feature extraction and construction using genetic programming for rotating machinery fault diagnosis. *IEEE Transactions on Cybernetics*, *51*(10), 4909–4923. https://doi.org/10.1109/TCYB.2020.3032945

Peng, H., Ying, C., Tan, S., Hu, B., & Sun, Z. (2018). An improved feature selection algorithm based on ant colony optimization. *IEEE Access*, *6*, 69203–69209. https://doi.org/10.1109/ACCESS.2018.2879583

Pevny, T., Bas, P., & Fridrich, J. (2010). Steganalysis by subtractive pixel adjacency matrix. *IEEE Transactions on Information Forensics and Security*, *5*, 215–224. https://doi.org/10.1109/TIFS.2010.2045842

Pourpanah, F., Shi, Y., Lim, C. P., Hao, Q., & Tan, C. J. (2019). Feature selection based on brain storm optimization for data classification. *Applied Soft Computing*, *80*, 761–775. https://doi.org/10.1016/j.asoc.2019.04.037

Quinlan, J. R. (2014). *C4. 5: Programs for machine learning*. Elsevier.

Rajamohana, S., & Umamaheswari, K. (2018). Hybrid approach of improved binary particle swarm optimization and shuffled frog leaping for feature selection. *Computers & Electrical Engineering*, *67*, 497–508. https://doi.org/10.1016/j.compeleceng.2018.02.015

Rao, H., Shi, X., Rodrigue, A. K., Feng, J., Xia, Y., Elhoseny, M., Yuan, X., & Gu, L. (2019). Feature selection based on artificial bee colony and gradient boosting decision tree. *Applied Soft Computing*, *74*, 634–642. https://doi.org/10.1016/j.asoc.2018.10.036

Rashedi, E., Nezamabadi-Pour, H., & Saryazdi, S. (2009). GSA: A gravitational search algorithm. *Information Sciences*, *179*, 2232–2248. https://doi.org/10.1016/j.ins.2009.03.004

Rashedi, E., Nezamabadi-Pour, H., & Saryazdi, S. (2010). BGSA: Binary gravitational search algorithm. *Natural Computing*, *9*, 727–745. https://doi.org/10.1007/s11047-009-9175-3

Remeseiro, B., & Bolon-Canedo, V. (2019). A review of feature selection methods in medical applications. *Computers in Biology and Medicine*, *112*, 103375. https://doi.org/10.1016/j.compbiomed.2019.103375

Robinson, D., Chen, Q., Xue, B., Killeen, D., Fraser-Miller, S., Gordon, K. C., Oey, I., & Zhang, M. (2021). Genetic algorithm for feature and latent variable selection for nutrient assessment in horticultural products. *2021 IEEE Congress on Evolutionary Computation (CEC)*. IEEE.

Rostami, M., Berahmand, K., Nasiri, E., & Forouzande, S. (2021). Review of swarm intelligence-based feature selection methods. *Engineering Applications of Artificial Intelligence*, *100*, 104210. https://doi.org/10.1016/j.engappai.2021.104210

Rupasinghe, T., Gentle, I. E., Beilharz, T. H., Pettolino, F. A., Djordjevic, J. T., Lo, T. L., Uwamahoro, N., Rupasinghe, T., Tull, D. L., McConville, M., Beaurepaire, C., Nantel, A., Lithgow, T., Mitchell, A. P., & Traven, A. (2010). Cell wall integrity is linked to mitochondria and phospholipid homeostasis in Candida albicans through the activity of the post-transcriptional regulator Ccr4-Pop2. *Molecular Microbiology*, *79*(4), 968–989. https://doi.org/10.1111/j.1365-2958.2010.07503.x

Sadeghian, Z., Akbari, E., & Nematzadeh, H. (2021). A hybrid feature selection method based on information theory and binary butterfly optimization algorithm. *Engineering Applications of Artificial Intelligence*, *97*, 104079. https://doi.org/10.1016/j.engappai.2020.104079

Salcedo-Sanz, S., Delser, J., Landa-Torres, I., Gil-López, S., & Portilla-Figueras, J. (2014). The coral reefs optimization algorithm: A novel metaheuristic for efficiently solving optimization problems. *Scientific World Journal*, *2014*. https://doi.org/10.1155/2014/739768

Saremi, S., Mirjalili, S., & Lewis, A. (2017). Grasshopper optimisation algorithm: Theory and application. *Advances in Engineering Software*, *105*, 30–47. https://doi.org/10.1016/j.advengsoft.2017.01.004

Sawhney, R., Mathur, P., & Shankar, R. (2018). A firefly algorithm based wrapper-penalty feature selection method for cancer diagnosis. International Conference on Computational Science and Its Applications, Melbourne, VIC, Australia, (pp. 438–449). Springer.

Sayed, G. I., Hassanien, A. E., & Azar, A. T. (2019). Feature selection via a novel chaotic crow search algorithm. *Neural Computing & Applications*, *31*, 171–188. https://doi.org/10.1007/s00521-017-2988-6

Sayed, G. I., Tharwat, A., & Hassanien, A. E. (2019). Chaotic dragonfly algorithm: An improved metaheuristic algorithm for feature selection. *Applied Intelligence*, *49*, 188–205. https://doi.org/10.1007/s10489-018-1261-8

Selvakumar, B., & Muneeswaran, K. (2019). Firefly algorithm based feature selection for network intrusion detection. *Computers & Security*, *81*, 148–155. https://doi.org/10.1016/j.cose.2018.11.005

Sharma, M., & Kaur, P. (2021). A comprehensive analysis of nature-inspired meta-heuristic techniques for feature selection problem. *Archives of Computational Methods in Engineering*, *28*, 1103–1127. https://doi.org/10.1007/s11831-020-09412-6

Sharma, P., Sundaram, S., Sharma, M., Sharma, A., & Gupta, D. (2019). Diagnosis of Parkinson's disease using modified grey wolf optimization. *Cognitive Systems Research*, *54*, 100–115. https://doi.org/10.1016/j.cogsys.2018.12.002

Shi, Y. (2011). Brain storm optimization algorithm. International conference in swarm intelligence, Chongqing, China, (pp. 303–309). Springer.

Singh, N., & Singh, S. (2017). Hybrid algorithm of particle swarm optimization and grey wolf optimizer for improving convergence performance. *Journal of Applied Mathematics*, *2017*. https://doi.org/10.1155/2017/2030489

Song, Q., Ni, J., & Wang, G. (2011). A fast clustering-based feature subset selection algorithm for high-dimensional data. *IEEE Transactions on Knowledge and Data Engineering*, *25*(1), 1–14. https://doi.org/10.1109/TKDE.2011.181

Storn, R., & Price, K. (1997). Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, *11*, 341–359. https://doi.org/10.1023/A:1008202821328

Suthaharan, S. (2016). Machine Learning Models and Algorithms for Big Data Classification. *Integrated Series in Information Systems* (1st ed., Vol. 36, pp. 359). Springer. ISBN: 978-1-4899-7641-3. https://doi.org/10.1007/978-1-4899-7641-3

Tadist, K., Mrabti, F., Nikolov, N. S., Zahi, A., & Najah, S. (2021). SDPSO: Spark distributed PSO-based approach for feature selection and cancer disease prognosis. *Journal of Big Data*, *8*, 1–22. https://doi.org/10.1186/s40537-021-00409-x

Taradeh, M., Mafarja, M., Heidari, A. A., Faris, H., Aljarah, I., Mirjalili, S., & Fujita, H. (2019). An evolutionary gravitational search-based feature selection. *Information Sciences*, *497*, 219–239. https://doi.org/10.1016/j.ins.2019.05.038

Tran, B., Xue, B., & Zhang, M. (2017). A new representation in PSO for discretization-based feature selection. *IEEE Transactions on Cybernetics*, *48*(6), 1733–1746. https://doi.org/10.1109/TCYB.2017.2714145

Tran, B., Xue, B., & Zhang, M. (2019). Genetic programming for multiple-feature construction on high-dimensional classification. *Pattern Recognition*, *93*, 404–417. https://doi.org/10.1016/j.patcog.2019.05.006

Tran, B., Zhang, M., & Xue, B. (2016). A PSO based hybrid feature selection algorithm for high-dimensional classification. 2016 IEEE congress on evolutionary computation (CEC), Vancouver, BC, Canada, (pp. 3801–3808). IEEE.

Tu, Q., Chen, X., & Liu, X. (2019). Multi-strategy ensemble grey wolf optimizer and its application to feature selection. *Applied Soft Computing*, *76*, 16–30. https://doi.org/10.1016/j.asoc.2018.11.047

Tubishat, M., Abushariah, M. A., Idris, N., & Aljarah, I. (2019). Improved whale optimization algorithm for feature selection in Arabic sentiment analysis. *Applied Intelligence*, *49*, 1688–1707. https://doi.org/10.1007/s10489-018-1334-8

Tubishat, M., Ja'afar, S., Alswaitti, M., Mirjalili, S., Idris, N., Ismail, M. A., & Omar, M. S. (2021). Dynamic salp swarm algorithm for feature selection. *Expert Systems with Applications*, *164*, 113873. https://doi.org/10.1016/j.eswa.2020.113873

Wan, Y., Wang, M., Ye, Z., & Lai, X. (2016). A feature selection method based on modified binary coded ant colony optimization algorithm. *Applied Soft Computing*, *49*, 248–258. https://doi.org/10.1016/j.asoc.2016.08.011

Wang, G. -G., Guo, L., Gandomoi, A. H., Hao, G. -S., & Wang, H. (2014). Chaotic krill herd algorithm. *Information Sciences*, *274*, 17–34. https://doi.org/10.1016/j.ins.2014.02.123

Wang, M., Wu, C., Wang, L., Xiang, D., & Huang, X. (2019). A feature selection approach for hyperspectral image based on modified ant lion optimizer. *Knowledge-Based Systems*, *168*, 39–48. https://doi.org/10.1016/j.knosys.2018.12.031

Wang, Y., Wang, J., Liao, H., & Chen, H. (2017). Unsupervised feature selection based on Markov blanket and particle swarm optimization. *Journal of Systems Engineering and Electronics*, *28*, 151–161. https://doi.org/10.21629/JSEE.2017.01.17

Wolpert, D. H., & Macready, W. G. (1997). No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, *1*, 67–82. https://doi.org/10.1109/4235.585893

Xu, H., Xue, B., & Zhang, M. (2020). A duplication analysis-based evolutionary algorithm for biobjective feature selection. *IEEE Transactions on Evolutionary Computation*, *25*(2), 205–218. https://doi.org/10.1109/TEVC.2020.3016049

Xue, B., & Chen, G. (2016). *Guest editorial: Special issue on evolutionary optimization, feature reduction and learning*. Springer.

Xue, B., & Zhang, M. (2020). Evolutionary computation for feature selection and feature construction. GECCO '20: Proceedings of the 2020 Genetic and Evolutionary Computation Conference Companion, Cancún, Mexico, (pp. 1283–1312). https://doi.org/10.1145/3377929.3389857

Xue, B., Zhang, M., Browne, W. N., & Yao, X. (2015). A survey on evolutionary computation approaches to feature selection. *IEEE Transactions on Evolutionary Computation*, *20*(4), 606–626. https://doi.org/10.1109/TEVC.2015.2504420

Xue, Y., Xue, B., & Zhang, M. (2019). Self-adaptive particle swarm optimization for large-scale feature selection in classification. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, *13*, 1–27. https://doi.org/10.1145/3340848

Yan, C., Ma, J., Luo, H., & Patel, A. (2019). Hybrid binary coral reefs optimization algorithm with simulated annealing for feature selection in high-dimensional biomedical datasets. *Chemometrics and Intelligent Laboratory Systems*, *184*, 102–111. https://doi.org/10.1016/j.chemolab.2018.11.010

Yang, Xin-She. (2010). Firefly Algorithm, Lévy Flights and Global Optimization. Research and Development in Intelligent Systems XXVI, United Kingdom, (pp. 209–218). Springer. https://doi.org/10.1007/978-1-84882-983-1_15

Yang, X. -S., & Deb, S. (2009). *Cuckoo search via Lévy flights. World congress on nature & biologically inspired computing (NaBIC)*. IEEE.

Yu, H., Wang, G., Yang, D., & Wu, Z. (2002). Knowledge reduction algorithms based on rough set and conditional information entropy. Data Mining and Knowledge Discovery: Theory, Tools, and Technology IV, Orlando, FL, USA, (pp. 422–431).

Zakeri, A., & Hokmabadi, A. (2019). Efficient feature selection method using real-valued grasshopper optimization algorithm. *Expert Systems with Applications*, *119*, 61–72. https://doi.org/10.1016/j.eswa.2018.10.021

Zawbaa, H. M., Emary, E., Grosan, C., & Bongard, J. (2016). Feature selection via chaotic antlion optimization. *Plos One*, *11*(3), e0150652. https://doi.org/10.1371/journal.pone.0150652

Zhang, L., Mistry, K., Lim, C. P., & Neoh, S. C. (2018). Feature selection using firefly optimization for classification and regression models. *Decision Support Systems*, *106*, 64–85. https://doi.org/10.1016/j.dss.2017.12.001

Zhang, X., Xu, Y., Yu, C., Heidari, A. A., Li, S., Chen, H., & Li, C. (2020). Gaussian mutational chaotic fruit fly-built optimization and feature selection. *Expert Systems with Applications*, *141*, 112976. https://doi.org/10.1016/j.eswa.2019.112976

Zhang, Y., Li, H. -G., Wang, Q., & Peng, C. (2019). A filter-based bare-bone particle swarm optimization algorithm for unsupervised feature selection. *Applied Intelligence*, *49*, 2889–2898. https://doi.org/10.1007/s10489-019-01420-9

Zhang, Z., Cheng, Y., & Liu, N. C. (2014). Comparison of the effect of mean-based method and z-score for field normalization of citations at the level of web of science subject categories. *Scientometrics*, *101*, 1679–1693. https://doi.org/10.1007/s11192-014-1294-7

Zheng, Y., Li, Y., Wang, G., Chen, Y., Xu, Q., Fan, J., & Cui, X. (2018). A novel hybrid algorithm for feature selection based on whale optimization algorithm. *IEEE Access*, *7*, 14908–14923. https://doi.org/10.1109/ACCESS.2018.2879848

Gokulnath C Babu and Shantharajah S P. (2019). An optimized feature selection based on genetic approach and support vector machine for heart disease. *Cluster Computer*, *22*(S6), 14777–14787. https://doi.org/10.1007/s10586-018-2416-4