Optimizing Feature Selection Based on the Black-Winged Kite Algorithm

Mohanad Ridha Ghanim*[ID], Farah Neamah Abbas[ID], Rasha Thamer Shawi[ID]

College of Education, Computer Science Department, Mustansiriyah University, Baghdad 10064, Iraq

Corresponding Author Email: muhannadridha@uomustansiriyah.edu.iq

ABSTRACT

Feature selection is a critical phase in machine learning, aimed at improving model performance by pinpointing the most relevant elements within a dataset. However, it encounters difficulties, especially with high-dimensional data, such as the possibility of being ensnared in local optima and the considerable computational expense of navigating an extensive feature space. This work introduces a novel feature selection approach using the Black-Winged Kite Algorithm (BKA) to address these challenges efficiently. The BKA methodology integrates two novel methods. Initially, it employs a probability-based initialization method to enhance the correlation between features and labels, facilitating expedited convergence in the optimization process. Secondly, it utilizes a task formation method predicated on feature correlation, segmenting the process into two tasks: the primary job chooses highly linked features, while the secondary task finds non-redundant ones. Utilizing a multi-task transfer mechanism, the algorithm disseminates information across tasks, enhancing search space exploration and diminishing the probability of local optima. Experiments on five high-dimensional datasets demonstrate that this BKA-based approach attains superior classification accuracy with a reduced number of features and exhibits quicker performance than conventional approaches, establishing it as an effective solution for high-dimensional data issues.

1. INTRODUCTION

Classification is one of the most common problems in machine learning, where a model aims to predict an outcome for any future data. Feature selection is considered one of the most important preprocessing approaches to improve classification performance by eliminating redundant variables and retaining only the most relevant variables [1]. FS has been applied so far with promising results in several domains, such as job scheduling [2], text categorization [3], picture processing [4], disease diagnosis [5], and gene selection [6]. This increases exponentially in the number of features n of a dataset and therefore gives rise to a combinatorial problem. This exponential growth of the search space has been referred to as "curse of dimensionality." This obviously causes considerable computational problems for the usual feature selection approaches. Hence, feature selection methods need to balance two necessities: effective exploration of the solution space, on one hand, and keeping computational costs reasonable, on the other hand.

A variety of efficient feature selection (FS) methods have been developed to tackle the issues posed by high-dimensional data. Prominent instances include feature selection methodologies using surrogate models [7], multi-objective feature selection utilizing differential evolution [8], cluster-guided particle swarm optimization (PSO) for imbalanced datasets with absent values [9], and variable-length PSO-based approaches [10]. Feature selection approaches may be classified into three primary categories: filter-based, wrapper-based, and embedded-based, based on their relationship with statistical or machine learning techniques. The filter-based approach identifies feature subsets by assessing intrinsic metrics, like distance, correlation, or information gain among features [11]. This method is computationally efficient since it excludes categorization models, hence decreasing time and computational expenses. Nonetheless, its drawback lies in the somewhat decreased categorization accuracy relative to other methods.

The wrapper-based method evaluates feature subsets using exact classifiers, such as support vector machines (SVM), decision trees (DT), K-nearest neighbors (KNN), or artificial neural networks (ANN) [12]. This strategy often yields more accuracy; nonetheless, it entails increased computational expenses due to the need of continuously training the classifier on diverse feature subsets. To balance the advantages and disadvantages of these two techniques, researchers have suggested the embedded-based feature selection approach. This approach integrates feature selection inside the training phase of the learning model, using the classifier to evaluate the importance of each feature. Embedded strategies often exhibit more efficiency than wrapper-based methods, lowering computational costs; nonetheless, their effectiveness depends on the classifier used [13]. This dependence may limit the generalizability of the results, since the importance of selected parameters is influenced by the model's characteristics.

In high-dimensional feature selection jobs, an effective

search method is crucial for enhancing the selection process. Evolutionary algorithms are often used because to their strong global search capabilities. Common algorithms include genetic programming (GP) [14], ant colony optimization (ACO) [15], genetic algorithms (GAs) [16], and particle swarm optimization (PSO) [17]. Particle Swarm Optimization (PSO) is particularly favored in feature selection because to its superior global search capabilities, lower computational requirements, and simple implementation compared to other evolutionary computing (EC) techniques. The effectiveness of PSO in feature selection has been extensively shown in previous researches [18, 19]. For instance, the study [20] introduced a variable-length PSO-based feature selection method that enables particles to have shorter and varied lengths, hence reducing the search space and enhancing efficiency. Similarly, the study [21] presented a PSO-based methodology that prioritizes the selection of features of intermediate importance, recognizing their capacity to improve population development. Surrogate models boost PSO performance by approximating fitness values and reducing the frequency of costly fitness checks. Despite the advantages of PSO, several PSO-based feature selection techniques are prone to local optima, particularly in high-dimensional data contexts. This is mostly due to the vast number of features, which may limit the algorithm's ability to comprehensively explore the whole feature space.

Evolutionary multitasking (EMT) has been a significant emphasis in optimization, facilitating the simultaneous resolution of several problems within a unified evolutionary framework. This approach improves global convergence by facilitating information sharing across tasks [22]. EMT has been effectively used in several optimization scenarios, such as parameter extraction [23] and vehicle routing [24], because to its robust global search abilities and efficient inter-task knowledge transfer mechanisms [25-27]. An essential mechanism in EMT for enabling information transfer is assortative mating, in which the random mating probability (rmp) governs the extent of cross-task evolution [28]. By adjusting the RMP, researchers may equalize the intensity of information transmission across related activities, therefore enhancing the overall optimization process.

Pan et al. [29] presented a hybrid feature selection method using multifactor particle swarm optimization (MFPSO) to address high-dimensional classification challenges. This approach utilizes knowledge transfer across tasks to improve classification accuracy and reduce search time. By using EMT principles, particularly its cross-task evolutionary processes, MFPSO enhances the effectiveness of feature selection in high-dimensional spaces, becoming it a crucial tool for tackling complex classification problems.

The variety and complexity of optimization projects mean that not all assignments benefit from knowledge transfer, thus impeding the optimization process. Knowledge transfer across similar or complementary roles positively influences the optimization process [30]. When activities have similar search regions or objectives, they are more likely to provide advantageous information for each other. Task selection algorithms are often categorized into two types: similarity-based and feedback-based [28]. The similarity-based approach focuses on identifying activities with comparable search areas or objectives, hence enabling knowledge transfer across inherently related tasks. The feedback-driven method evaluates tasks after each iteration and finds those that might enhance the optimization process based on the progress made.

This technology enables adaptive adjustments throughout the optimization process, ensuring that information is sent only when it is likely to improve outcomes. Both methods aim to ensure that only beneficial information is distributed, obstructing the transmission of extraneous or detrimental knowledge across unrelated tasks.

This study introduces an innovative dual-task feature selection technique grounded on the BKA, which may improve classification accuracy by using a reduced subset of features. The specific contributions of this study are as follows:
•　Proposes an evolutionary feature selection method using multitasking for knowledge sharing in high-dimensional data.
•　Introduces population initialization based on feature probability using the maximum information coefficient (MIC) and random initialization for diversity.
•　Implements an inflection point selection strategy to divide features into frontier and non-redundant sets for separate task search spaces.
•　Develops a knowledge transfer strategy using global optimal location crossing with transfer intensity to guide knowledge sharing.
•　Optimizes the search process with an acceleration function to balance early exploration and late convergence for high-quality solutions.

This document is organized as follows: Section 2 delineates pertinent literature. Section 3 delineates the recommended methodology. Section 4 delineates the experimental setup, while Section 4 examines the experimental outcomes and analysis. Ultimately, Section 5 presents the paper's conclusion.

2. RELATED WORK

2.1 Feature selection theory

Feature selection is the process of discovering and choosing a subset of relevant characteristics for model development. Its objective is to enhance model performance by removing redundant or unnecessary data, mitigating overfitting, improving generalization, and decreasing computing expenses. Techniques including filter methods, wrapper methods, and embedded methods [4, 9]. Feature selection involves selecting a subset of characteristics from the original set to enhance the value of the goal function. In the classification problem, assume a dataset of M samples and D features, where feature selection is used to pick K features (K < D) from the original feature set to minimize the classification error rate F. Thus, the whole feature selection problem may be expressed as equation 1, where xi=1 indicates that the feature is selected; otherwise, it is not chosen.

$$\begin{cases} minF(X) \\ s.tX = (x1, x2, \ldots, xn), xn = (0,1), \forall i \in \{1, 2, \ldots, D\} \end{cases} \quad (1)$$

Feature selection gets progressively more difficult in high-dimensional datasets, where the quantity of features (D) may significantly surpass the number of samples (M). Such datasets often demonstrate sparsity, elevated noise levels, and intricate feature interactions, which might impair the efficacy of conventional feature selection techniques. Filter-based approaches may neglect feature interactions, but wrapper and

embedding methods might be computationally intensive owing to their repetitive processes. These problems highlight the need for innovative and effective feature selection methods designed for high-dimensional contexts.

This study aims to address these challenges by leveraging the BKA to optimize feature selection in high-dimensional datasets. By focusing on both accuracy and computational efficiency, this work highlights the importance of adaptive and robust feature selection approaches to tackle the growing complexity of modern data.

2.2 Literature survey of feature selection

This section provides a succinct summary of current research focused on feature selection methodologies. A recent work [31] introduces a novel multiobjective approach called Multi-Objective Relative Discriminative Criterion (MORDC) for feature selection in datasets. MORDC aims to balance the minimization of redundant features with the augmentation of relevance to the target class. A new work [32] introduced a distinctive multi-label feature selection method termed MLACO, grounded in Ant Colony Optimization. MLACO aims to identify the most advantageous features inside the feature space by evaluating both relevance and redundancy variables. In the study [33], a sequence of filtering approaches is systematically used on synthetic data characterized by variations in the number of prominent features, the degree of output noise, the interactions between features, and a progressive rise in sample size. Information Gain is a notable approach for feature selection [34]. Information Gain involves evaluating the importance of attributes in data by analyzing their contribution to reducing uncertainty throughout the classification process. It is used in several research initiatives [35].

The particle swarm optimization (PSO) method has been widely used in the problem of feature selection. A novel method amalgamates genetic algorithms with particle swarm optimization (PSO) to ascertain the ideal set of attributes [17]. The main goal is to streamline and accelerate the finding of effective methods for feature selection from large datasets. In a distinct application of PSO, Qu et al. [36] provided an innovative approach for feature selection. It refines the problem by producing a set of salient qualities. This carefully selected set of attributes is effective in improving text classification performance while reducing execution time. In a hybrid PSO algorithm including an intelligent learning mechanism [17], researchers use a self-learning strategy to provide optimum exploration possibilities. Concurrently, they use a competitive learning-based prediction method to enhance the algorithm's application of existing information. This is used to achieve a balance between pursuing new possibilities and using existing ones.

Ant Colony Optimization (ACO) has been widely used in the feature selection domain. In a recent application of Ant Colony Optimization (ACO) [15], researchers presented a novel approach for text feature selection using a wrapper technique, which is coupled with ACO to guide the feature selection process. Additionally, it utilizes KNN as a classifier to evaluate and generate a candidate subset of optimum features. The feature subset outcomes, obtained from the suggested ACO-KNN method, were used as input to identify and extract the necessary features. In the study [37], the Ant Colony Optimization (ACO) for feature selection utilizes a hybrid search approach that combines the benefits of both

wrapper and filter approaches. To enable this mixed study, they set new laws and evaluated heuristic data. Concurrently, ants are guided along precise paths while creating subsets of the network, with a specified graph included at each phase of the method.

2.3 BKA

The BKA is an optimization algorithm derived by mimicking the hunting and scouting behaviors of Black-Winged Kites (BWK) [38]. In BKA, each kite represents a potential solution to the problem, and all kites together form a set of candidate solutions. In a D-dimensional search space, at generation t, the current position vector of a kite can be represented as $X_i^t = [X_{i1}^t, X_{i2}^t, \dots, X_{iD}^t]$, and the velocity vector as $V_i^t = [V_{i1}^t, V_{i2}^t, \dots, V_{iD}^t]$. The search process is guided by two positions: the individual best position of each kite $gbest_i^t = [gbest_{i1}^t, gbest_{i2}^t, \dots, gbest_{iD}^t]$ and the global best position $gbest_i^t = [gbest_1^t, gbest_2^t, \dots, gbest_D^t]$ among all kites up to the current generation [39]. The kites update their positions and velocities according to the following equations:

$$V_{id}^{t+1} = \omega * V_{id}^t + c_1 * r_1 * (gbest_{id}^t - X_{id}^t) + c_1 * r_1 * (gbest_{id}^t - X_{id}^t) \qquad (2)$$

$$X_{id}^{t+1} = X_{id}^t + V_{id}^{t+1} \qquad (3)$$

In this context, t denotes the current iteration number, ω signifies the inertia weight, c_1 and c_2 represent the acceleration coefficients for cognitive and social factors, respectively, while r_1 and r_2 are random values uniformly distributed between 0 and 1.

2.4 Max information coefficient

Mutual information is a statistic in information theory that quantifies the extent of dependency between two correlated random variables [40]. Mutual information quantifies the extent to which knowledge of one random variable, X or Y, diminishes the uncertainty associated with the other variable. Mutual information is characterized as:

$$I(X, Y) = \sum_{y \in Y} \sum_{x \in X} p(x, y) \, log \frac{p(x, y)}{p(x)p(y)} \qquad (4)$$

In this context, $p(x)$ and $p(y)$ represent the marginal probabilities of the variables x and y, respectively, whereas $p(x,y)$ denotes their joint probability. The Maximum Information Coefficient (MIC) is a non-parametric technique grounded on mutual information, applicable to both linear and nonlinear data relationships. MIC is determined by dividing the data space into grids and assessing the mutual information across various grid resolutions, as specified by:

$$MIC(x, y) = max_{|X||Y|<B} \frac{I(X, Y)}{log_2(\min(|X||Y|))} \qquad (5)$$

where, B is the upper limit on the number of grid cells, usually set to the power of 0.6 of the data size.

2.5 Evolutionary multitasking optimization

Evolutionary Multitasking Optimization (EMTO) [41] is an

innovative method in evolutionary optimization designed to improve global search efficacy via the exchange of information across various activities throughout the evolutionary process. The mathematical representation of multitasking optimization is:

$$\{\{X_1^*, X_2^*, \dots, X_K^*\}\} = \{argminf_1(x_1), argminf_2(x_2), \dots, argminf_K(x_K)\} \tag{6}$$

where, X_K^* denotes the optimal solution for task K. Multifactorial Optimization (MFO) is a specific form of EMTO, where multiple tasks are assigned to separate search spaces. This approach allows for simultaneous handling of multiple tasks, which may operate either independently or interactively. MFO aims to uncover useful knowledge through parallel searches, treating each task as a factor influencing the overall evolution process.

EMTO and MFO are optimization approaches that enable the exchange of genetic material (solutions) across jobs. Each task is regarded as a unique optimization problem, and the evolutionary method is designed to facilitate this interaction. The concept is that tasks may include intrinsic similarities, and insights gained from one work may aid in resolving another. MFO, a specific form of EMTO, assigns several tasks to distinct search regions, seeing each task as a separate factor influencing the overall evolutionary process. The main idea is to do concurrent searches across several tasks, enabling the discovery of pertinent information that can be shared and used to improve the optimization process. The primary advantage of EMTO and MFO is in their ability to proficiently tackle complex, high-dimensional optimization problems by using synergies across tasks. This makes them particularly suitable for scenarios necessitating the concurrent resolution of many interconnected problems, such as feature selection in high-dimensional datasets or multi-objective optimization.

3. PROPOSED FRAMEWORK

Enhancing feature selection with the BWK algorithm entails emulating the predatory behavior of BWK. The program begins with a collection of solutions (kites) and iteratively adjusts their placements according to exploration and exploitation methodologies. Kites identify subsets of ideal characteristics by balancing exploration (global search) with exploitation (local search). The method assesses feature subsets using a fitness function reliant on classification accuracy and several performance measures that will be elaborated upon subsequently. Through iterations, the kites converge on the most relevant feature subsets, therefore lowering dimensionality and improving model performance. The versatility and resilience of BWK make it efficient in intricate feature selection tasks across several fields.

The BKA is a nature-inspired optimization method that emulates the predatory behavior of BWK to address feature selection challenges.

3.1 Initialization in BKA

The initialization phase in the BKA is critical as it establishes the foundation for the optimization process. Let the dataset D be defined as $D=\{X,y\}$, where X is the feature matrix and y is the target class vector. The feature matrix $X \in R^{n \times m}$, consists of n samples and mmm features, each represented as $X=\{x_1, x_2, \dots, x_m\}$. The primary goal of the BWK algorithm in feature selection is to determine the optimal feature subset $X_s \subseteq X$, that maximizes model performance while minimizing the number of selected features.

During this phase, a population of N agents, designated as BWK, is started, with each agent symbolizing a possible feature subset represented as a binary vector $V \in \{0,1\}m$. Each bit in the vector represents a feature, with 1 indicating inclusion and 0 indicating exclusion of the feature. The starting population is produced randomly to provide diversity across the feature subsets inside the feature space. Mathematically, the initial position V_i^0 of each agent i at iteration $t=0$ is defined as:

$$V_i^0 = \{v_{i1}^0, v_{i2}^0, \dots, v_{im}^0\}, v_{ij}^0 \in \{0,1\} \tag{7}$$

The initialization entails specifying essential parameters, including the population size N, the maximum number of iterations T, and the fitness function $f(V)$, often grounded in classification accuracy or an alternative model performance indicator. This stage guarantees that the algorithm starts with a varied array of candidate solutions for effective exploration and exploitation of the feature space in later rounds.

3.2 Fitness evaluation in BWK

The fitness of each agent in the population is evaluated based on the performance of the selected feature subset. Let $f(V_i)$ denote the fitness function of agent i, where V_i represents the feature subset selected by that agent. The objective of the fitness function is to optimize the balance between minimizing the number of selected features and maximizing the classification performance of the model. The fitness function can be mathematically formulated as:

$$f(V_i) = \alpha . A(V_i) - \beta . \frac{|V_i|}{m} \tag{8}$$

where,
- $A(V_i)$ is the accuracy (or any performance metric) of the classifier trained on the selected features V_i,
- $|V_i|$ represents the number of features in the selected subset,
- m is the total number of features in the dataset,
- α and β are weight parameters controlling the trade-off between classification accuracy and the size of the feature subset.

Each agent i encodes its feature subset as a binary vector $V_i \in \{0,1\}^m$, where 1 indicates that the corresponding feature is selected, and 0 indicates that it is excluded. For each agent, the selected feature subset $X_s \subseteq X$ is used to train a classification model. The performance of the model, usually measured in terms of accuracy, precision, or F1-score, is then used to calculate the fitness score $f(V_i)$.

In reality, the parameters α and β may be ascertained by empirical calibration to get the optimal equilibrium between precision and feature subset magnitude. Commence by establishing $\alpha = 1$ and $\beta = 0.5$, thereafter modifying according to the empirical findings. In balanced settings, a common selection is $\alpha = 1$ and $\beta = 1$, which equally prioritizes accuracy and feature reduction. If accuracy is deemed twice as significant as feature reduction, then $\alpha = 2$ and $\beta = 1$ are established. If feature reduction is prioritized, then $\alpha = 1$ and $\beta = 2$ are used.

3.3 Exploration in BWK (third phase)

In the third phase of the BKA, the emphasis shifts to exploration, when the agents (kites) traverse the feature space to discern potential subsets of features. The exploration phase is essential for averting early convergence to local optima and guaranteeing that the algorithm covers a wide area of the solution space. Let V_i^t represent the location of the i-th agent (feature subset) at iteration t. Agents are prompted to advance into uncharted regions of the search space by applying random perturbations to their existing locations.

The movement of each agent during exploration is modeled by updating its feature subset V_i^t using a probabilistic approach. The new position V_i^{t+1} is generated based on the current position and a random factor that allows the agent to explore new feature combinations. Mathematically, the position update for each agent is defined as:

$$V_i^{t+1} = V_i^t + \gamma . R \qquad (9)$$

where,

- γ is the exploration factor, controlling the intensity of the random search,
- R is a random vector with elements drawn from a uniform distribution $R \in \{-1,0,1\}^m$, which introduces random changes to the current feature subset.

To ensure that the updated feature subset remains valid (i.e., binary values), the new position V_i^{t+1} is typically normalized using a sigmoid or threshold function:

$$V_i^{t+1} = \begin{cases} 1 \ if \ sigmoid \ (V_i^{t+1}) > threshold \\ 0 \ otherwise. \end{cases} \qquad (10)$$

This transformation guarantees that each agent maintains a binary feature selection vector, with 1 denoting feature selection and 0 signifying exclusion.

The exploration phase enables agents to extensively investigate the solution space, uncovering a variety of feature subsets. By creating novel feature combinations, the algorithm circumvents local minima and sets the stage for the next exploitation phase, during which more promising areas of the search space will be optimized.

3.4 Exploitation in BWK (fourth phase)

In the fourth phase of the BKA, attention transitions from exploration to exploitation, as agents hone their search in attractive regions of the feature space. Exploitation guarantees that agents use the most advantageous feature subsets discovered during the exploration phase. This approach entails optimizing the agents' placements (i.e., chosen feature subsets) to enhance their fitness ratings. The objective of exploitation is to get an optimum or near-optimal collection of features that improves classification efficacy.

Let V_i^t be the position of the i-th agent at iteration t, and let V_{best}^t represent the position of the best-performing agent in the population at the same iteration. During exploitation, agents adjust their positions to move closer to V_{best}^t, thereby focusing on refining feature subsets that have demonstrated superior performance. The position update rule for exploitation can be expressed as:

$$V_i^{t+1} = V_i^t + \eta . (V_{best}^t - V_i^t) + \delta . \mathrm{N}(0,1) \qquad (11)$$

where,

- η is the exploitation factor that controls the convergence speed toward the best solution,
- V_{best}^t is the position of the best-performing agent in the population,
- N(0,1) is a Gaussian random variable that introduces small stochastic perturbations for local search,
- δ is a scaling factor for the Gaussian noise.

The exploitation process encourages agents to refine their feature subsets by learning from the best solutions discovered so far, while small perturbations allow agents to explore nearby solutions. The updated position V_i^{t+1} is again normalized to maintain a binary vector, ensuring that only valid feature subsets are considered. In addition to moving toward the best agent, a subset of agents may also exchange information with neighboring agents in the population. This cooperative method enables agents to exchange insights on optimal feature subsets, hence expediting convergence. The exploitation phase guarantees that agents concentrate on enhancing the quality of their feature subsets via incremental modifications, thereby refining the solutions discovered during the exploration phase. The equilibrium between exploitation and exploration is essential for the efficacy of the BWK algorithm, as it guarantees a comprehensive search of the solution space and convergence to an optimum solution.

3.5 Population update in BWK

The fifth step in the BKA involves population updating, where agents update their positions (feature subsets) based on both individual performance and the collective knowledge of the population. This step consolidates the results of the exploration and exploitation phases by adjusting the positions of agents (i.e., their selected feature subsets) to reflect improvements. The goal is to guide agents toward better solutions while maintaining diversity in the population to prevent premature convergence. Let V_i^{t+1} represent the updated position of the i-th agent (feature subset) at iteration t+1. The fitness of each agent is re-evaluated using the same fitness function $f(V_i)$ as described earlier. Based on the new fitness values, the population is updated using the following strategies:

1. **Selection of the Best Agent**: The agent with the highest fitness score V_{best}^{t+1} is identified. This agent serves as a reference for guiding the movement of other agents in subsequent iterations.

$$V_{best}^{t+1} = \arg \max f(V_i^{t+1}), i = 1,2, \dots , N \qquad (12)$$

2. **Agent Position Update**: Each agent adjusts its position based on both its own current position and the position of V_{best}^{t+1}. The position update equation is the same as in the exploitation step:

$$V_i^{t+1} = V_i^t + \eta . (V_{best}^t - V_i^t) + \delta . \mathrm{N}(0,1) \qquad (13)$$

This update ensures that each agent moves closer to the best solution while also introducing minor stochastic variations to maintain diversity in the population.

3. **Maintaining Population Diversity**: To avoid premature convergence to suboptimal solutions, a diversity mechanism is introduced. Agents that are too close to V_{best}^{t+1} may be perturbed to explore new regions of the search space. This can be achieved by

randomly flipping some bits in the binary vector representation of their feature subset V_i^{t+1}:

$$V_{ij}^{t+2} = \begin{cases} 1 - V_{ij}^{t+1} \ with \ probability \ p_{flip} \\ V_{ij}^{t+1} \ otherwise. \end{cases} \qquad (14)$$

Here, p_{flip} is a small probability that ensures controlled randomness, allowing agents to explore new feature subsets without completely losing focus on the best solutions.

4. **Replacement of Poor Solutions**: Agents exhibiting significantly low fitness ratings compared to the population average may be replaced by newly initiated random agents. This stage facilitates the introduction of new possible solutions into the population and augments global search capacity.

The algorithm repeats Steps 3 through 5 until one of the following stopping criteria is met:

Maximum number of iterations T is attained. Convergence: The disparity between the fitness values of successive ideal solutions falls within a specified threshold for a designated number of iterations, suggesting that the algorithm has probably reached an optimal solution.

This population update step ensures that the BWK algorithm continues refining the search for an optimal feature subset, balancing between improvement of current solutions and exploration of new possibilities. Once any of the stopping criteria is met, the algorithm terminates, and the best feature subset V_{best}^t found up to the current iteration is returned as the final solution. This feature subset represents the most optimal balance between minimizing the number of selected features and maximizing the classification performance according to the fitness function.

4. EXPERIMENTAL DESIGN

The experimental approach entails assessing the efficacy of the BKA in optimizing feature selection using benchmark datasets, juxtaposing classification accuracy and computing efficiency with other feature selection techniques.

4.1 Dataset study

This research employs five high-dimensional datasets, available at http://featureselection.asu.edu. Table 1 presents a comprehensive overview of the datasets, including the quantity of samples, characteristics, and classes. A characteristic common to all datasets is their elevated feature-to-sample ratio, including several attributes with a limited sample size. The imbalance poses a considerable difficulty for precise classification, as the elevated dimensionality amplifies task complexity, while the restricted sample size limits the data available for training and validation. Thus, these datasets are optimal for assessing feature selection techniques under demanding circumstances.

To enhance comprehension of the datasets and their relevance to the proposed method, we performed statistical studies on the characteristics and samples within each dataset. All datasets have continuous numerical features that denote gene expression levels or other high-dimensional biological data, with features assessed on a comparable scale, making them appropriate for direct comparison without significant adjustment. The numerical ranges differ among datasets: the

11 Tumor dataset features range from 0 to 1,000, predominantly clustering between 0 and 100, signifying sparse high-intensity signals; the DLBCL dataset features span 0 to 800, with a median around 150, indicating moderate variability; the Brain Tumor 2 dataset displays highly skewed numerical ranges, peaking near 900 with most values below 200, reflecting heterogeneous data distributions; the Lung Cancer 2 dataset presents a more uniform distribution, with values ranging from 0 to 1,200, denoting diverse expression levels; and the Leukemia 3 dataset showcases a broad range from 0 to 1,500, with a dense concentration of features beneath 200. Initial analysis indicates significant inter-feature correlation across all datasets, with redundancy ratios over 30%, highlighting the need for efficient feature selection techniques to reduce unnecessary or duplicated features. Class imbalance is apparent, particularly in the DLBCL and Leukemia 3 datasets, where certain classes predominate in the sample distribution, underscoring the need for strong assessment criteria to address unbalanced data. Moreover, the feature-to-sample ratios for these datasets surpass 60:1 in every instance, with the Brain Tumor 2 dataset exhibiting the highest ratio of 207:1. This presents dimensionality challenges that intensify the risks of overfitting and computational difficulties, highlighting the necessity of effective feature selection.

Table 1. Fundamental information of five datasets

Dataset	Class	Sample	Features
11 Tumor	11	174	12533
DLBCL	2	77	5469
Brain Tumor 2	4	50	10367
Lung Cancer2	5	203	12600
Leukemia 3	3	72	11225

4.2 Classification accuracy

Classification accuracy is a crucial parameter for evaluating the effectiveness of feature selection methods in high-dimensional classification problems. Enhanced classification accuracy signifies that certain attributes are more advantageous for the classification task. This study compares the suggested method with two widely used intelligent optimization algorithms for feature selection, namely PSO and GA, which have been employed by researchers over the last five years to illustrate its superiority. Refer to the results table for more information.

Table 2. Mean and standard deviation of classification accuracy (mean ± standard deviation) obtained by the intelligent optimization and BWK-FS algorithms over 25 independent runs on the 5 datasets

Dataset	PSO	GA	BWK-FS
11 Tumor	83.44±3.22(+)	87.57±2.64(+)	93.24±2.71
DLBCL	92.65±3.12(+)	94.28±3.02(+)	98.13±1.01
Brain Tumor 2	84.76±2.42(+)	85.48±3.23(+)	91.85±4.32
Lung Cancer2	88.85±4.10(+)	93.33±5.34(+)	96.98±2.11
Leukemia 3	93.96±2.88(+)	91.44±2.92(+)	99.22±0.55

Table 2 displays the mean and standard deviation of classification accuracy across five datasets, using three distinct algorithms: Particle Swarm Optimization (PSO), Genetic Algorithm (GA), and Black-Winged Kite-based Feature Selection (BWK-FS). The results are derived from 25 separate

trials, illustrating the reliability and efficacy of each method. In every dataset, BWK-FS surpasses both PSO and GA regarding classification accuracy, exhibiting markedly higher mean values and smaller standard deviations, which signifies its resilience and enhanced optimization proficiency. In the 11 Tumor dataset, BWK-FS attains a mean accuracy of 93.24%, exceeding PSO and GA by about 10%. In the DLBCL dataset, BWK-FS achieves an accuracy of 98.13%, surpassing GA by over 4%. This indicates that BWK-FS proficiently identifies essential elements to enhance classification efficacy.

The standard deviation of BWK-FS is consistently lower than or equivalent to that of other algorithms, notably evident in the Leukemia 3 dataset, where BWK-FS exhibits a deviation of just 0.55, compared to GA's 2.92. This signifies a more consistent performance over several executions, which is essential for guaranteeing dependability in practical applications. The outcomes indicated by "(+)" for PSO and GA imply their comparatively competitive but inferior performance relative to BWK-FS. BWK-FS exhibits substantial improvements in classification accuracy across all datasets, validating its efficacy as an intelligent optimization technique in feature selection endeavors.

BWK-FS is a strong and dependable feature selection approach that demonstrates reduced standard deviation values across all datasets, signifying its dependability and strength. The reduced standard deviation signifies the variability in accuracy across 25 separate trials, demonstrating enhanced consistency in performance. In the 11 Tumor Dataset, BWK-FS attains a standard deviation of 2.71, which is lower than that of PSO (3.22) and similar to GA (2.64), indicating its stability despite the dataset's large dimensionality. In the DLBCL Dataset, BWK-FS has the lowest standard deviation (1.01) relative to PSO (3.12) and GA (3.02), indicating its resilience in managing lower sample numbers. The stability of BWK-FS in the Leukemia 3 Dataset is notably shown by its standard deviation of just 0.55, which is much lower than that of PSO (2.88) and GA (2.92). The combination of elevated mean accuracy and minimal standard deviation illustrates BWK-FS's efficacy as an intelligent optimization technique for feature selection, guaranteeing consistent performance throughout several iterations, hence making it a more dependable option than PSO and GA.

To facilitate comprehension, Figure 1 presents histograms of feature distributions for each dataset, emphasizing the numerical ranges and skewness of the features. These visualizations depict the variety and sparsity within the datasets, which are essential factors for assessing feature selection approaches.
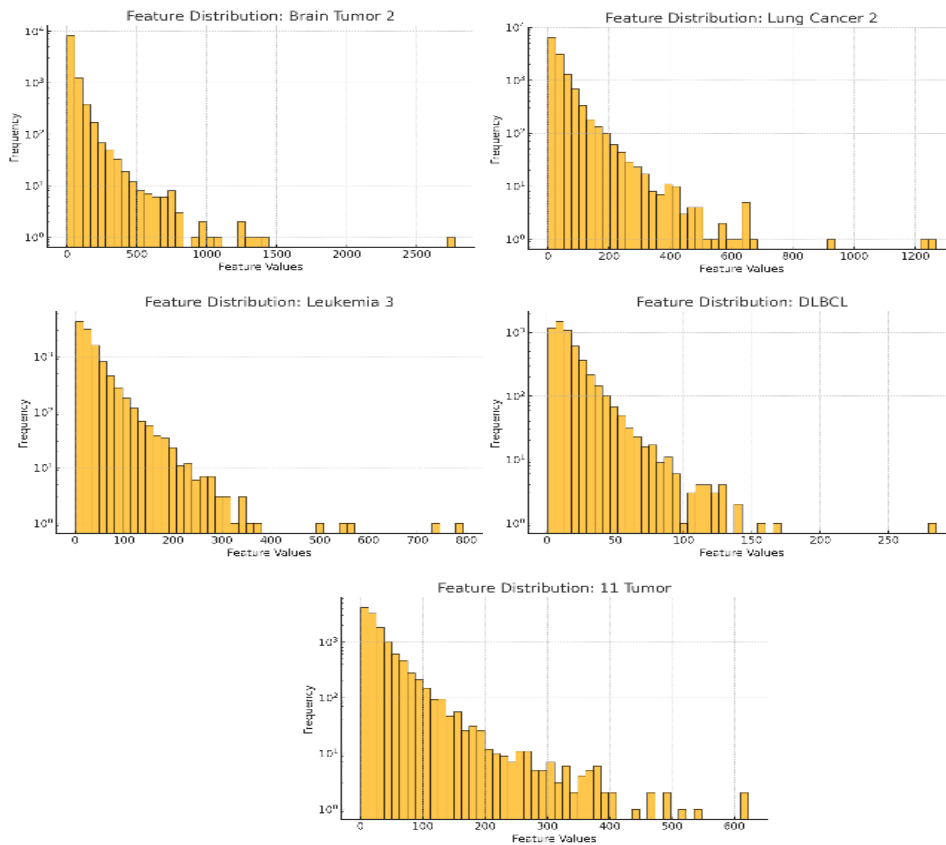

Figure 1. Provide histograms of feature distributions for each dataset

4.3 Training time

Training time of the algorithm is one of the critical factors in assessing the effectiveness of any feature selection approach. When compared over five datasets, the training time for the proposed optimization method is better than that of the other algorithms. Table 3 examines the number of selected features through the work of three different optimization algorithms, including PSO, GA, and BWK-FS in five different datasets. This table helps to highlight how well an individual method could reduce feature space while directly having impacts on the reduction of computational complexity and raising the model's performances.

In fact, among all datasets, the significant fewer feature choices made by BWK-FS compared to those by PSO and GA in this experiment suggest the better feature reduction capability of BWK-FS. For example, BWK-FS selected 289 features on 11 Tumor but PSO and GA choose 6234 and 6003,

respectively. This huge reduction indicates that BWK-FS can identify the most informative features for classification without performance degradation, which can also be seen from the accuracy values in Table 2 formula. This trend is the same on the rest of the datasets, for example, on the DLBCL dataset, BWK-FS identifies only 165 features against 2195 and 2415 selected by PSO and GA, respectively. This indicates that BWK-FS has higher efficiency on many data sets with much better reductions on feature sets.

Table 3. Quantity of features picked by the smart optimization and BWK-FS methods across the five datasets

Dataset	PSO	GA	BWK-FS
11 Tumor	6234	6003	289
DLBCL	2195	2415	165
Brain Tumor 2	4256	4450	285
Lung Cancer2	6034	5878	245
Leukemia 3	3222	5250	346

This reduction drastically cuts overfitting risk and boosts computational efficiency, thus allowing for faster training and testing times. This feature selection scheme keeps or even improves the accuracy of classification, observed from Table 2. Therefore, the BWK-FS will be powerful machinery in intelligent optimization applied to feature selection tasks, as this method allows it to find a small group of very informative features.

Table 4 describes the training time in seconds taken by PSO, GA, and BWK-FS for five datasets. The table depicts the computational efficiency of each technique, which is a very important feature in real-world applications, especially when large datasets are involved or high-speed processing is required. BWK-FS has always shown the minimum training time compared to PSO and GA in all datasets. On the 11 Tumor dataset, the training of BWK-FS was completed in 824.2 seconds, far faster than PSO at 1945.22 seconds and GA at 1588.42 seconds. The reduction in training time is probably due to BWK-FS selecting fewer features—as shown in Table 3—reducing the computational burden of the model.

Table 4. The training duration used by two intelligent optimization methods and BWK-FS across five datasets (unit: seconds).

Dataset	PSO	GA	BWK-FS
11 Tumor	1945.22	1588.42	824.2
DLBCL	107.25	217.85	78.46
Brain Tumor 2	328.36	227.89	172.4
Lung Cancer2	1928.41	2210.93	858.93
Leukemia 3	398.67	275.24	89.02

On the Lung Cancer2 dataset, the training time taken by BWK-FS is 858.93 s, which is less than half compared to PSO (1928.41 s) and GA (2210.93 s). For smaller datasets such as DLBCL, BWK-FS completes its training in less than 78.46 s, whereas PSO and GA take approximately 107.25 s and 217.85 s, respectively. The performance of BWK-FS is very impressive on those datasets where GA and PSO take a longer training time. It is due to the reduction of computational complexity along with the capability of BWK-FS for selecting fewer features that enhanced its performance in terms of training time. Thus, BWK-FS is not only more accurate, as depicted from Table 2, but it is also much faster, which creates a trade-off between the accuracy and computational efficiency for the feature selection tasks.

The ability of BWK-FS to select fewer but more relevant features can be attributed to two main factors: the algorithmic mechanism and problem characteristics.

- **Focused Exploration and Exploitation**: The BWK-FS algorithm draws inspiration from the foraging behavior of BWK, effectively balancing global exploration with local exploitation. This mechanism guarantees that the search process discovers the most promising feature subsets while avoiding needless investigation of irrelevant domains.
- **Adaptive Fitness Function**: The fitness function in BWK-FS aims to maximize two conflicting objectives: enhancing classification accuracy and reducing the number of chosen characteristics. This dual-objective methodology intrinsically penalizes the use of irrelevant or redundant characteristics, prompting the algorithm to concentrate on more compact, impactful subsets.
- **Feature Interdependency Handling**: BWK-FS incorporates inter-agent communication mechanisms that facilitate the exchange of information among agents to effectively capture and manage the complex interdependencies between features, thereby enhancing the overall feature selection process and improving model performance.

Tables 2 and 3 detail the performance metrics of the algorithms across five datasets. Notably, BWK-FS achieves the highest classification accuracy in all datasets, with a peak accuracy of 98.13% in the DLBCL dataset, outperforming PSO and GA by substantial margins. For example, in the 11 Tumor dataset, BWK-FS attains 93.24% accuracy compared to 83.44% and 87.57% by PSO and GA, respectively. Table 4 presents the number of features selected by each algorithm. BWK-FS identifies significantly fewer features, selecting only 289 in the 11 Tumor dataset compared to 6234 and 6003 for PSO and GA, respectively. Similar reductions are observed across other datasets, including DLBCL, where BWK-FS selects 165 features versus 2195 and 2415 for PSO and GA. The efficiency of BWK-FS in reducing training time is evident in Table 4. For the 11 Tumor dataset, BWK-FS requires only 824.2 seconds to train the model, considerably less than the 1945.22 and 1588.42 seconds required by PSO and GA, respectively. Similar trends are observed in datasets like DLBCL and Leukemia 3, where BWK-FS demonstrates faster model training.

5. CONCLUSION

Relative effectiveness and efficiency of the three optimization techniques—Particle Swarm Optimization (PSO), Genetic Algorithm (GA), and Black-Winged Kite-based Feature Selection (BWK-FS)—are shown via comparative study. The BWK-FS routinely exceeds PSO and GA in terms of classification accuracy, therefore displaying strong and consistent performance across several datasets. This highlights its potential to maintain stability across many runs and enhance categorization results. Furthermore, very beneficial is BWK-FS in significantly reducing the dimensionality of the feature collection. Using fewer but more important features helps BWK-FS lower overfitting in addition to improve model interpretability. Reduced dimensionality directly lowers computer complexity by enabling quicker training durations and preservation of suitable classification accuracy. By besting the trade-off

between model complexity and predictive performance, BWK-FS offers usually a robust option for feature selection. In high-dimensional data processing, its remarkable accuracy with less features and faster training cycles makes it a useful tool.

Notwithstanding its robust performance, BWK-FS has limits that need more examination. Although it efficiently diminishes features in moderately high-dimensional datasets, its scalability for very high-dimensional data (e.g., over 100,000 features) has yet to be evaluated, necessitating more investigation. Moreover, BWK-FS exhibits sensitivity to hyperparameters such as α and β inside its fitness function, indicating a need for automated or adaptive tuning methodologies to improve usability. It may also encounter difficulties with significantly skewed datasets, when accuracy alone is inadequate; integrating measures such as F1-score or MCC might enhance performance. While evaluated on biological data, its applicability to fields such as text mining, image processing, or finance necessitates more assessment and possible modifications. Finally, the initialization step, which encompasses candidate solution creation, may be computationally expensive for big datasets, underscoring the need for more efficient methods to mitigate overhead. Mitigating these constraints might substantially improve the application and performance of BWK-FS.

In conclusion, BWK-FS is the most successful algorithm of the three, offering enhanced classification accuracy, a notable decrease in the number of chosen features, and expedited training periods across all datasets. The amalgamation of superior accuracy, feature efficiency, and computational velocity renders BWK-FS an exceptionally advantageous instrument for intelligent optimization in feature selection endeavors. Its capacity to equilibrate performance and efficiency establishes it as a formidable method for addressing intricate machine learning challenges.

REFERENCES

[1] Tijjani, S., Ab Wahab, M.N., Noor, M.H.M. (2024). An enhanced particle swarm optimization with position update for optimal feature selection. Expert Systems with Applications, 247: 123337. https://doi.org/10.1016/j.eswa.2024.123337

[2] Zhang, F.F., Mei, Y., Nguyen, S., Zhang, M.J. (2020). Evolving scheduling heuristics via genetic programming with feature selection in dynamic flexible job-shop scheduling. IEEE Transactions on Cybernetics, 51(4): 1797-1811. https://doi.org/10.1109/TCYB.2020.3024849

[3] Sinoara, R.A., Camacho-Collados, J., Rossi, R.G., Navigli, R., Rezende, S.O. (2019). Knowledge-enhanced document embeddings for text classification. Knowledge-Based Systems, 163: 955-971. https://doi.org/10.1016/j.knosys.2018.10.026

[4] Chen, Z.P., Tondi, B., Li, X.L., Ni, R.R., Zhao, Y., Barni, M. (2019). Secure detection of image manipulation by means of random feature selection. IEEE Transactions on Information Forensics and Security, 14(9): 2454-2469. https://doi.org/10.1109/TIFS.2019.2901826

[5] Ali, L., Zhu, C., Zhou, M.Y., Liu, Y.P. (2019). Early diagnosis of Parkinson's disease from multiple voice recordings by simultaneous sample and feature selection. Expert Systems with Applications, 137: 22-28. https://doi.org/10.1016/j.eswa.2019.06.052

[6] Gil, F., Osowski, S. (2020). Feature selection methods in gene recognition problem. In 2020 IEEE 21st International Conference on Computational Problems of Electrical Engineering (CPEE) (Online Conference), Poland, pp. 1-4. https://doi.org/10.1109/CPEE50798.2020.9238726

[7] Zhang, N., Gupta, A., Chen, Z.F., Ong, Y.S. (2021). Evolutionary machine learning with minions: A case study in feature selection. IEEE Transactions on Evolutionary Computation, 26(1): 130-144. https://doi.org/10.1109/TEVC.2021.3099289

[8] Nag, K., Pal, N.R. (2019). Feature extraction and selection for parsimonious classifiers with multiobjective genetic programming. IEEE Transactions on Evolutionary Computation, 24(3): 454-466. https://doi.org/10.1109/TEVC.2019.2927526

[9] Chen, K., Xue, B., Zhang, M.J., Zhou, F.Y. (2020). An evolutionary multitasking-based feature selection method for high-dimensional classification. IEEE Transactions on Cybernetics, 52(7): 7172-7186. https://doi.org/10.1109/TCYB.2020.3042243

[10] Bayati, H., Dowlatshahi, M.B., Paniri, M. (2020). MLPSO: A filter multi-label feature selection based on particle swarm optimization. In 2020 25th International Computer Conference, Computer Society of Iran (CSICC) Tehran, Iran, pp. 1-6. https://doi.org/10.1109/CSICC49403.2020.9050087

[11] Tran, B., Xue, B., Zhang, M.J. (2018). Variable-length particle swarm optimization for feature selection on high-dimensional classification. IEEE Transactions on Evolutionary Computation, 23(3): 473-487. https://doi.org/10.1109/TEVC.2018.2869405

[12] Got, A., Zouache, D., Moussaoui, A., Abualigah, L., Alsayat, A. (2024). Improved manta ray foraging optimizer-based SVM for feature selection problems: A medical case study. Journal of Bionic Engineering, 21(1): 409-425. https://doi.org/10.1007/s42235-023-00436-9

[13] Theng, D., Bhoyar, K.K. (2024). Feature selection techniques for machine learning: A survey of more than two decades of research. Knowledge and Information Systems, 66(3): 1575-1637. https://doi.org/10.1007/s10115-023-02010-5

[14] Devarriya, D., Gulati, C., Mansharamani, V., Sakalle, A., Bhardwaj, A. (2020). Unbalanced breast cancer data classification using novel fitness functions in genetic programming. Expert Systems with Applications, 140: 112866. https://doi.org/10.1016/j.eswa.2019.112866

[15] Verma, G., Sahu, T.P. (2024). TOPSIS-ACO based feature selection for multi-label classification. International Journal of Computers and Applications, 1-18. https://doi.org/10.1080/1206212X.2024.2321683

[16] Maleki, N., Zeinali, Y., Niaki, S.T.A. (2021). A k-NN method for lung cancer prognosis with the use of a genetic algorithm for feature selection. Expert Systems with Applications, 164: 113981. https://doi.org/10.1016/j.eswa.2020.113981

[17] Bayuaji, L., Amzah, M.Y., Pebrianti, D. (2024). Optimization of feature selection in support vector machines (SVM) using recursive feature elimination (RFE) and particle swarm optimization (PSO) for heart disease detection. In 2024 9th International Conference on Mechatronics Engineering (ICOM), Kuala Lumpur, Malaysia, pp. 304-309.

https://doi.org/10.1109/ICOM61675.2024.10652561

[18] 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

[19] Hu, Y., Zhang, Y., Gong, D. (2020). Multiobjective particle swarm optimization for feature selection with fuzzy cost. IEEE Transactions on Cybernetics, 51(2): 874-888. https://doi.org/10.1109/TCYB.2020.3015756

[20] Paul, D., Jain, A., Saha, S., Mathew, J. (2021). Multi-objective PSO based online feature selection for multi-label classification. Knowledge-Based Systems, 222: 106966. https://doi.org/10.1016/j.knosys.2021.106966

[21] Chen, K., Xue, B., Zhang, M., Zhou, F. (2021). Correlation-guided updating strategy for feature selection in classification with surrogate-assisted particle swarm optimization. IEEE Transactions on Evolutionary Computation, 26(5): 1015-1029. https://doi.org/10.1109/TEVC.2021.3134804

[22] Liu, P., Xu, B., Xu, W. (2024). A new evolutionary multitasking algorithm for high-dimensional feature selection, IEEE Access, 12. https://doi.org/10.1109/ACCESS.2024.3418809

[23] Bali, M., Murthy, P.V.R. (2020). Bio-molecular event extraction using classifier ensemble-of-ensemble. In Data Management, Analytics and Innovation: Proceedings of ICDMAI 2020, pp. 445-462. https://doi.org/10.1007/978-981-15-5619-7_32

[24] Zhang, F.F., Mei, Y., Nguyen, S., Zhang, M.G. (2020). A preliminary approach to evolutionary multitasking for dynamic flexible job shop scheduling via genetic programming. In Proceedings of the 2020 Genetic and Evolutionary Computation Conference Companion, pp. 107-108. https://doi.org/10.1145/3377929.3389934

[25] Ding, J.L., Yang, C.E., Jin, Y.C., Chai, T.Y. (2017). Generalized multitasking for evolutionary optimization of expensive problems. IEEE Transactions on Evolutionary Computation, 23(1): 44-58. https://doi.org/10.1109/TEVC.2017.2785351

[26] Feng, L., Huang, Y.X., Zhou, L., Zhong, J.H., Gupta, A., Tang, K., Tan, K.C. (2020). Explicit evolutionary multitasking for combinatorial optimization: A case study on capacitated vehicle routing problem. IEEE Transactions on Cybernetics, 51(6): 3143-3156. https://doi.org/10.1109/TCYB.2019.2962865

[27] Zhou, L., Feng, L., Tan, K. C., Zhong, J.H., Zhu, Z.X., Liu, K., Chen, C. (2020). Toward adaptive knowledge transfer in multifactorial evolutionary computation. IEEE Transactions on Cybernetics, 51(5): 2563-2576. https://doi.org/10.1109/TCYB.2020.2974100

[28] Wei, T.Y., Wang, S.B., Zhong, J.H., Liu, D., Zhang, J. (2021). A review on evolutionary multitask optimization: Trends and challenges. IEEE Transactions on Evolutionary Computation, 26(5): 941-960. https://doi.org/10.1109/TEVC.2021.3139437

[29] Pan, X.Y., Lei, M.Z., Sun, J., Wang, H., Ju, T., Bai, L. (2024). An evolutionary feature selection method based on probability-based initialized particle swarm optimization. International Journal of Machine Learning and Cybernetics, pp. 1-20.

[30] Jiang, Y., Zhan, Z.H., Tan, K.C., Zhang, J. (2022). A bi-objective knowledge transfer framework for evolutionary many-task optimization. IEEE Transactions on Evolutionary Computation, 27(5): 1514-1528. https://doi.org/10.1109/TEVC.2022.3210783

[31] Potharlanka, J.L. (2024). Feature importance feedback with Deep Q process in ensemble-based metaheuristic feature selection algorithms. Scientific Reports, 14(1): 2923. https://doi.org/10.1038/s41598-024-53141-w

[32] Kaya, C., Kilimci, Z.H., Uysal, M., Kaya, M. (2024). Migrating birds optimization-based feature selection for text classification. arXiv preprint arXiv:2401.10270. https://doi.org/10.48550/arXiv.2401.10270

[33] Rafie, A., Moradi, P., Ghaderzadeh, A. (2023). A multi-objective online streaming multi-label feature selection using mutual information. Expert Systems with Applications, 216: 119428. https://doi.org/10.1016/j.eswa.2022.119428

[34] Song, X.F., Zhang, Y., Guo, Y.N., Sun, X.Y., Wang, Y.L. (2020). Variable-size cooperative coevolutionary particle swarm optimization for feature selection on high-dimensional data. IEEE Transactions on Evolutionary Computation, 24(5): 882-895. https://doi.org/10.1109/TEVC.2020.2968743

[35] Htun, H.H., Biehl, M., Petkov, N. (2023). Survey of feature selection and extraction techniques for stock market prediction. Financial Innovation, 9(1): 26. https://doi.org/10.1186/s40854-022-00441-7

[36] Qu, L.T., He, W.B., Li, J.F., Zhang, H., Yang, C., Xie, B. (2023). Explicit and size-adaptive PSO-based feature selection for classification. Swarm and Evolutionary Computation, 77: 101249. https://doi.org/10.1016/j.swevo.2023.101249

[37] Karimi, F., Dowlatshahi, M.B., Hashemi, A. (2023). SemiACO: A semi-supervised feature selection based on ant colony optimization. Expert Systems with Applications, 214: 119130. https://doi.org/10.1016/j.eswa.2022.119130

[38] Wang, J., Wang, W.C., Hu, X.X., Qiu, L., Zang, H.F. (2024). Black-winged kite algorithm: A nature-inspired meta-heuristic for solving benchmark functions and engineering problems. Artificial Intelligence Review, 57(4): 98. https://doi.org/10.1007/s10462-024-10723-4

[39] Chen, K.H., Lin, W.L., Lin, S.M. (2022). Competition between the black-winged kite and Eurasian kestrel led to population turnover at a subtropical sympatric site. Journal of Avian Biology, 2022(10): e03040. https://doi.org/10.1111/jav.03040

[40] Abdel-Basset, M., Mohamed, R., Hezam, I.M., Sallam, K.M., Hameed, I.A. (2024). Parameters identification of photovoltaic models using Lambert W-function and Newton-Raphson method collaborated with AI-based optimization techniques: A comparative study. Expert Systems with Applications, 255: 124777. https://doi.org/10.1016/j.eswa.2024.124777

[41] Zhao, H., Ning, X.H., Liu, X.T., Wang, C., Liu, J. (2023). What makes evolutionary multi-task optimization better: A comprehensive survey. Applied Soft Computing, 145: 110545. https://doi.org/10.1016/j.asoc.2023.110545