International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Data Mining-Based K-Nearest Neighbor Technique for Multiclass Dataset Feature Selection and Classification

[1*]R. Senthamil Selvi, [2]K. Fathima Bibi

Submitted: 07/02/2024 **Revised**: 15/03/2024 **Accepted**: 21/03/2024

Abstract: Data analysis is used to extract useful information from small or large datasets and gain insights for future recommendations and decision-making. Predictive analytics is the application of data mining and machine learning techniques to make predictions. However, there are some areas for improvement in the previous algorithm, such as an optimal solution to the finite problem not being found and complicated dataset parameter selection. The previous paper, Hybrid feature selection-based Binary ACO (HFSBACO) [2], achieved 98.6%. Still, it had some difficulties; There are complex dataset stages, and prediction could be more efficient because this data requires a lot of time and resources. It is challenging to extract relevant information.

To overcome the issue, we proposed the Machine learning techniques used for Classification based on K-Nearest Neighbor (KNN) for predicting multi-dataset using features. Initially, input the Multi-dataset taken from the UCI repository. First, the Dataset was pre-trained to remove the irrelevant, missing, and noisy data. Before building the model, Feature Correlation Coefficients (FCC) between various dependent and independent features were analyzed to determine the strength of the relationship between each dependent and independent feature of the Dataset. Pre-processing data to split the train 70% and testing 30% of data for feature selection. The second stage is extracting the relevant data from the dataset-based Enhanced Binary Cuckoo Search with Ant colony optimization Algorithm (EBCS-ACO) for selecting the feature values based on its nearest feature threshold weights or values. ACO estimates the feature weights sequence order to be maintained using this algorithm. Before Classification, the K-fold cross-validation method for training and testing data metrics varies, as some ways consider iterative validation. For each sample, the quality measures were determined based on the Receiver Operating Characteristic (ROC) Curve analysis. The last step is detecting the Dataset using the K-Nearest Neighbor (KNN) algorithm and evaluating the result based on the training and testing data. Receiver operating characteristic curves serve to assess and compare classification models objectively. The classification model considers precision, recall, accuracy, f1-score, ROC, and time complexity for best prediction, which results in better accuracy and prediction rate than previous methods.

Keywords: Machine learning, Hybrid, training and testing, Dataset, features, K-fold cross-validation.

1. Introduction

Data mining is identifying patterns in vast volumes of data that are intriguing, obvious, concealed, new, and cooperative. Data mining is seen as a natural evolution of information technology. This is because the traditional methods could be more suitable for large-scale, high-dimensional, heterogeneous, and distributed data. Data can now be stored in various databases and information repositories.

Machine learning techniques are widely used to create robust models that predict class membership from unlabelled observations. These approaches can achieve excellent classification accuracy on datasets with complicated behaviours and various applications. Machine learning has the disadvantage of making models increasingly complex, challenging to interpret, and expensive to evaluate on massive datasets. Ideally, we want to construct models that are easy to build, inexpensive to analyze and provide users with data information.

Feature selection is used to select the suitable features contributing to the machine learning model's prediction capacity. Irrelevant features may reduce model accuracy and lead to overfitting. The solution to the dimensionality problem is feature selection, especially given that data is unstructured. Feature selection approaches have been applied in various datasets such as Glass -214 samples, Letter -20000 models, Vehicle -846 examples, Spambase-4601 samples, and Waveform -5000 samples.

[1]Research Scholar in Computer Science, Thanthai Periyar Govt. Arts & Science College (A), Affiliated to Bharathidasan University, Tiruchirappalli, Tamilnadu, India
E-mail: senthamil.behappy@gmail.com
[2]Assistant Professor in Computer Science, Thanthai Periyar Govt. Arts & Science College (A), Affiliated to Bharathidasan University, Tiruchirappalli, Tamilnadu, India
E-mail: kfatima72@gmail.com

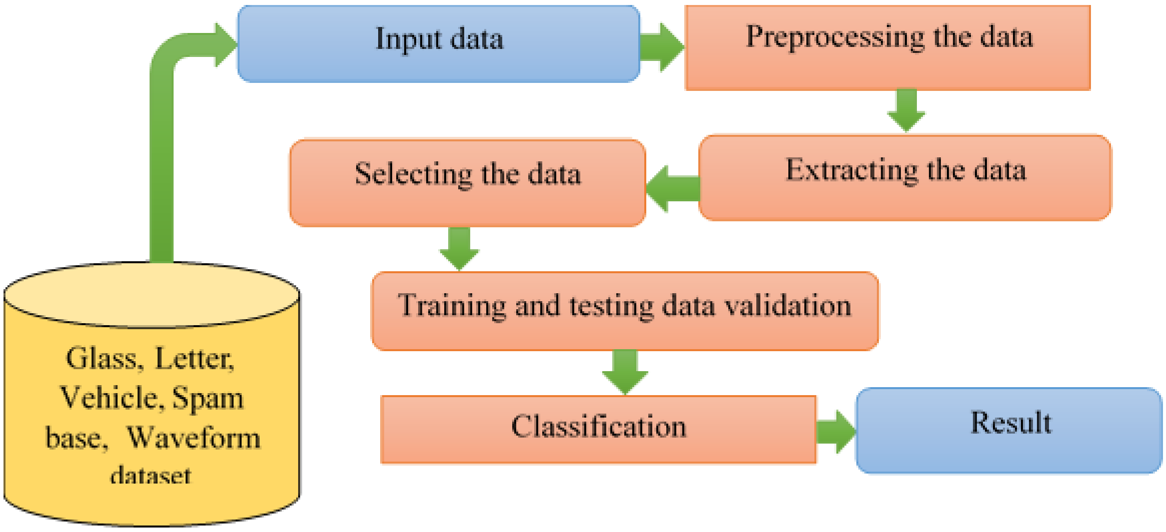


Figure 1: Dataset Feature selection flow diagram

Figure 1 describes Feature selection and classification results on a multiclass dataset using the proposed algorithm. The first step is to collect the Dataset from the standard repository and the multiclass Dataset for pre-processing to reduce unbalanced data values. Next, the features are extracted from the initial step, and the training, testing, and validating data are selected. It finally classifies the result based on the proposed method comparing the previous approaches.

The proposed method, the K-Nearest Neighbor (KNN) algorithm, is a robust machine learning algorithm applied in various applications. Because of their improved performance, KNN-based feature selection algorithms are gaining prominence. This has become the EBCS-ACO feature selection algorithm in bioinformatics and text classification. The KNN feature selection algorithm ranks feature based on their relative contribution to the Classification and eliminates many irrelevant characteristics. This method is repeated until a certain amount of features are chosen. KNN provides a good workflow for recursive feature selection, outperforms KNN feature selection techniques, and is more prone to noise and possible outliers.

A measure for determining the degree of correlation and redundancy between candidate qualities and classes. Use the Binary Cuckoo Search strategy with KNN to acquire the best feature set. On the other hand, our solution is adaptable regarding the learning algorithms that can be utilized in the process. This Dataset has one or more classes (asymmetric classes) with significantly more samples than the data from other classes. It is difficult to determine the characteristics that are appropriate for each class. This is because feature selection presents attributes that only reflect most classes.

KNN is trained on a single class dataset and requires no additional class datasets. For one class, the training time is solely determined by the amount of the Dataset. The EBCS-ACO method is used for feature selection and employs a sequential backward selection process. The suggested algorithm's training time is faster than previous multiclass classification methods, and its accuracy is comparable to other approaches. Hence, it is considered suitable for multiclass Classification.

2. Related work

Data classification is a fundamental task in data mining. It helps to organize data and extract useful information [1]. One of the most essential positions in multilabel learning is feature extraction. The multilabel Classification's performance can be significantly enhanced by lowering the multilabel Dataset's dimensionality. Research on multilabel feature extraction has garnered a lot of attention, and while significant progress has been made, there is still room for improvement. Typically, conventional multilabel feature extraction techniques are sensitive to noise and outliers [3]. However, sophisticated causal feature selection techniques usually work on a single dataset. Applying this procedure directly to different datasets may result in inconsistent outcomes, as the datasets may have other distributions [4]. Even though the convolutional brain organization can learn advanced semantic-level features for object affirmation, they have cut-off points to RGB-D scene portrayal. One of the challenges is that it needs to be straightforward to learn multimodal highlights for RGB-D scene acknowledgement ideally [5].

Component determination aims to dispose of unwanted attributes and further develop order execution. These objectives are fundamentally unrelated, and a choice should be taken when there are compromises between them. A few examinations have been conducted, including determination troubles, yet they have forever been centred on a solitary objective [6]. In a few situations, clients are interested in further developing grouping execution and reducing the expense of highlights. This is alluded to as cost-based, including choice. Most contemporary element determination procedures, notwithstanding, treat this issue as a solitary objective streamlining issue. The underlying work on Multiobjective Particle Swarm Optimisation (PSO) for cost-based highlight choice issues [7].

Generally, we assume the label assignment is either finished or nearly finished. In any case, missing names and unlabelled information are far, comprehensive, and predictable in accurate applications because of manual explanation and the significant expense of mark vulnerability. By consolidating Laplacian regularization in a scanty component determination structure to protect neighborhood consistency in preparing information, unsupervised spectral feature selection (USFS) approaches can create interpretable and unmistakable outcomes [8]. Improper sensor pairing can result in various issues, including selectivity overlap, significant computational overhead, and poor performance [9]. Because of the multiple schools of thought, establishing feature thresholds in HDD datasets remains challenging [10]. Using population coefficients, the influence of the starting population was investigated. The binary multiobjective grey wolf optimization comprises 20 wolves selected using the population bootstrapping approach [11]. Although it is a multiobjective issue, most existing methodologies consider including the choice of a solitary objective enhancement issue. The Multiobjective Grey Wolf Optimization (MOGWO) was, as of late, proposed to take care of the multiobjective improvement issue [12] high-dimensional multiclass inequality problems. The high-dimensional multiclass disparity problem has posed a serious challenge to conventional classifiers to perform the classification task effectively between minority and majority classes. Many attempts have been made to solve the problem of high-dimensional datasets or class imbalance [13].

The highlight option has become a significant issue for a long time in deciding the most fitting elements for a given grouping issue. Various techniques have been developed to consider support vector machines (SVMs) in the choice cycle. Such methodologies are robust yet frequently perplexing and expensive [14]. Prescient examination applications in advanced education organizations have turned into a pressing need. Prescient examination utilizes progressed investigation, including AI applications, to convey excellent execution and significant data for understudies at all instruction degrees [15]. Precise cleanliness evaluation is expected for the solid working of perplexing extraction frameworks. Despite the broad utilization of customary AI procedures like Artificial Neural Networks (ANNs) and Backing Vector Machines (SVMs), state assessment plans given a solitary order model keep on showing poor multiclass grouping execution, dispersed [16].

Prescient examination utilizes progressed investigation, including applications, to convey top-notch execution and significant data for understudies at all schooling degrees [17]. Exact cleanliness appraisal is expected for the dependable working of intricate extraction frameworks. Regardless of the inescapable use of conventional AI procedures like Counterfeit Brain Organizations (ANNs) and Backing Vector Machines (SVMs), state assessment plans in light of a solitary order model keep showing poor multiclass characterization execution, scattered—the more prominent the uniqueness [18].

Most feature selection method research has focused on single features or complete feature subsets, ignoring the impact of feature correlation and redundancy within feature subsets on classification results [19]. A sampling approach has been provided to increase the performance of a recently developed feature generation engine (FGM) in selecting meaningful features for very high dimensional problems [20].

In filtering through striking elements, highlight determination is a superb technique for managing tremendous relationships in hyperspectral image information. Despite the high-layered and multi-pack search, customary multiobjective developmental improvement-based highlight choice techniques are wasteful in looking and affirming because of the haphazardness of the methodology and the uncertainty of the advanced process [21].

In any case, as the quantity of extricated highlights develops, so does the dimensionality of the component vector. Besides, multi-layered vectors dramatically increment the complexity of the arranging strategy [22]. A feature selection approach to software defect prediction is currently presented, which can potentially increase the performance of traditional defect prediction (also known as intra-project defect prediction, WPDP) [23].

Feature or quality assurance is a data pre-processing framework that considers and feeds the most significant information to pointers. It minimizes computing overhead while improving classifier accuracy [24]. Long vectors with repetitive and uncertain elements (similitude among Vehicle and non-vehicle pictures) are created utilizing Pig highlights, bringing about high misclassification rates and costly grouping [25].

However, some accessible algorithms neglect label correlations, resulting in poor classification performance. Furthermore, most multilabel neighborhood rough set (MNRS)--based feature selection algorithms can only handle a limited amount of multilabel data [26]. Standard feature selection methods may only perform well for particular applications if all feature information is available beforehand. Furthermore, streaming feature selection approaches are computationally feasible for ultra-high-dimensional data analysis, especially when correlation effects between features are considered [27].

Most enduring element determination techniques rank all highlights given a particular standard, and the best-evaluated highlights are picked for future grouping or clustering tasks. Because feature redundancy is neglected, the selected features are frequently linked with one another, resulting in poor performance [28]. However, most studies to date have concentrated on one of them while ignoring interactions. Some studies look at both of these features at the same time. Separating the two elements may result in poor HAR performance [29].

One of the improvements was a reduction in the number of tests run. Unfortunately, many feature selection algorithms do not consider testing costs, resulting in an average test at a high cost [30]. One of its goals is to lower the number of features in the original Dataset to increase the prediction model's performance. Due to the difficulties in identifying the most informative genes for cancer prediction, feature selection strategies for selecting essential and relevant characteristics from vast and complicated datasets have been proposed [31].

The Binary Cuckoo Search strategy is proposed in this paper as a component choice calculation driven by a memory-based system to save the most valuable highlights found by the ideal arrangement [32]. For homogeneous or heterogeneous characteristics, binary or multiclass label feature selection, classification algorithms and associated pre-processing processes are often conducted independently. There have been few attempts to do feature selection on datasets with heterogeneous multiclass characteristics [33]. The author provides Enhanced Binary ACO (ABACO), a feature selection technique based on Ant Colony Optimisation (ACO). Features are regarded as graph nodes and are entirely related to one another when building a graph model. Each node in this network has two sub-nodes, one for feature selection and one for de-selection. When ants visit all the characteristics, the ant colony algorithm chooses nodes [34].

Similarly, the author used the ACO method to find the most valuable features of web pages. They devoted NB and KNN approaches to categorizing web pages [35]. Like that, the ACO method is used for text feature extraction for hate speech categorization [36]. However, many machine learning models are sensitive to noise and may not be suitable for solving real-world problems [37]. The study utilized CSO and TSFS algorithms for the finest attributes of multiclass datasets [38, 39]. They enhance the accuracy, sensitivity, and specificity performance. However, the methods provide poor performance for multiclass Classification. Therefore, the article [40, 41] explored the CSA method for multiclass Classification and perfect results.

2.1 Problem factors

- Feature selection has become an important issue in recent decades to determine the best features for a specific classification problem.
- Previous classification models still need better multiclass classification ability, high variability and significant discrepancies.
- The existing multiobjective evolutionary optimization-based feature selection methods need more efficient search.
- Previous methods have primarily been based on feature selection problems.
- Existing multilabel highlight extraction calculations are, by and large, boisterous and delicate to exceptions.

3. Materials and method

The proposed K nearest neighbor (KNN) approach is widely employed in biological and therapeutic applications. It is an effective learning method for multiclass classification and feature selection simultaneously. When estimating parameters, the suggested KNN approach can enhance intra-class distance while increasing inter-class distance. It is appropriate for small sample sizes and issues like class imbalance, typical in many real-world applications. Furthermore, model-based feature selection approaches can detect highly linked characteristics at the same time, eliminating the multivariate issues generated by multiple tests (multiclass Classification and feature selection).

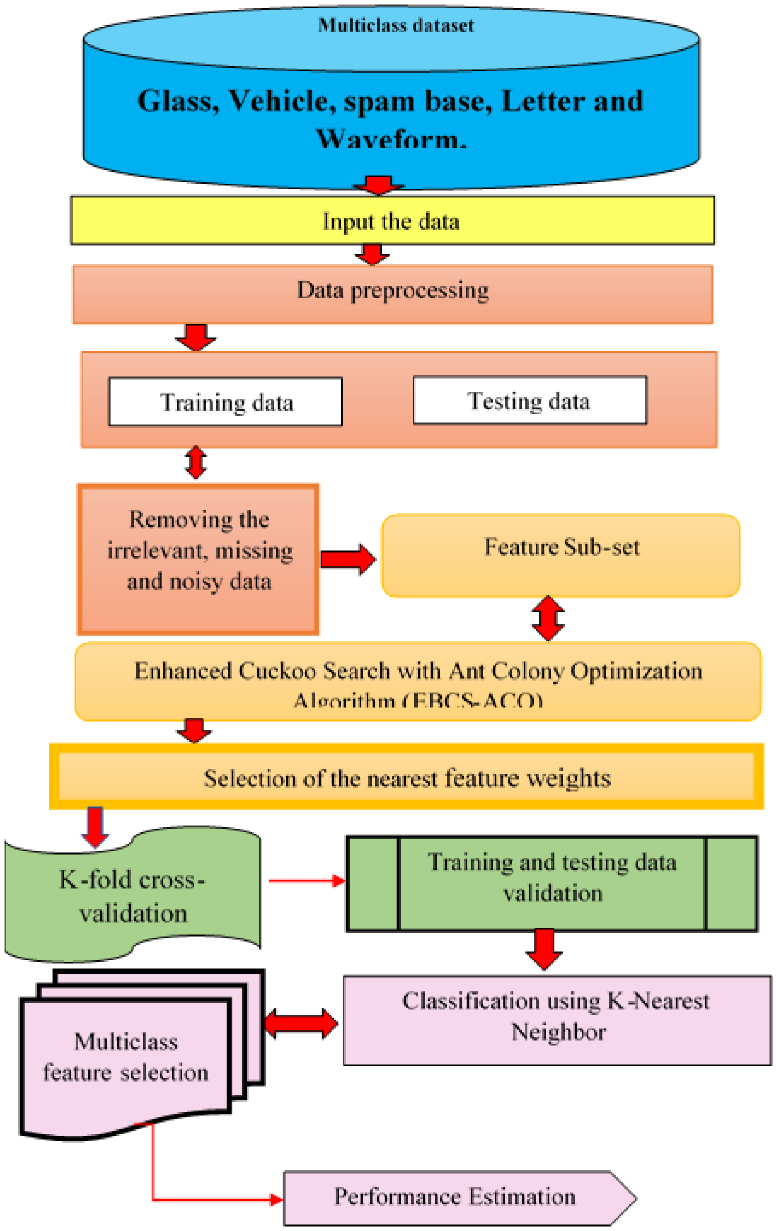


Figure2: Proposed block diagram

Figure 2 describes a proposed block diagram based on the K-nearest neighbor for classifying the results using multiclass datasets. Initially, we collected different data types in the standard repository for feature selection. Before feature selection, all data should enter the first stage of preprocessing. This step reduced the noise and irrelevant data from the Dataset using training and testing data. Then, the second phase is extracting the features from preprocess output data. It selects the maximum range or weight data values using Enhanced Binary Cuckoo Search with Ant colony optimization Algorithm (EBCS-ACO) to determine the features' nearest weights. Before Classification, the train and test data should be validated, so the K-fold validation method is used in this step. Finally, KNN classification predicts and estimates the performance for multiclass data feature selection for better performance.

3.1 Characteristics of Multiclass Datasets

Datasets contain many different data features that are used to train machine learning algorithms to find predictable designs through the Dataset.

Experimental results are run using five different datasets and a multi-dataset performance estimator from the UCI Machine Learning (ML) Repository. The accuracy of a dataset containing all instances is taken as a minimum certain for a multiclass performance estimator.

Table 1: Collection of Multi-dataset feature selection using ML approaches

Dataset	Features	Classes	Samples
Glass	9	6	214
Letter	16	26	20000
Vehicle	18	4	846
Spam base	57	2	4601
Waveform	40	3	5000

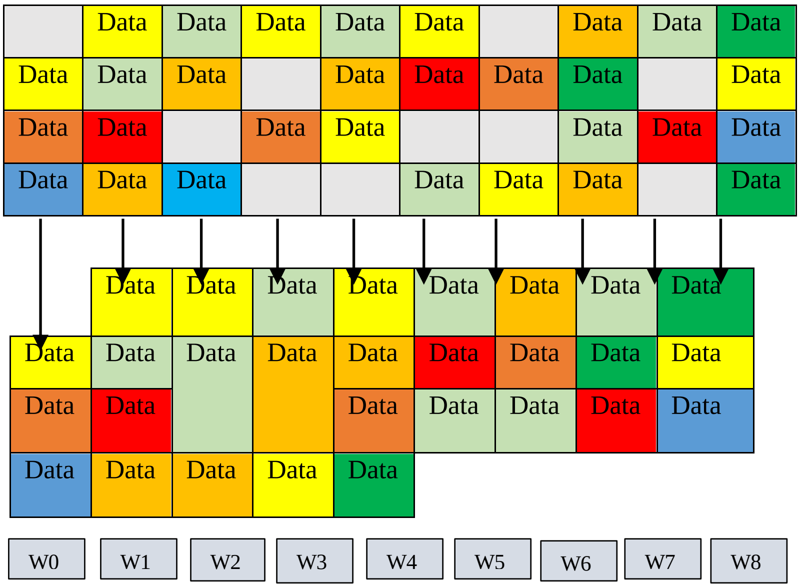
Table 1 describes the multi types of different Dataset features to Splitting the data for Training, Test and Validation Datasets in Machine Learning using below five datasets for feature selection as Glass -9-features, 6-class and 214 samples, Letter -16 features, 26 class and 20000 samples, Vehicle -18-features, 4-class and 846 samples, Spambase-57-features, 2-class and 4601 samples, Waveform - 40-features, 3-class and 5000 samples.

3.2 Data pre-processing

Data pre-processing can speed up the training/testing by correctly transforming and scaling the entire Dataset. Pre-processing the datasets to check for missing data must be in a format suitable for machine learning. It can clean data to remove noise and correct data inconsistencies. It pre-processes and filters accurate data that reduces noise, which is incomplete.

❖ **Handling missing data**

Data incompleteness is an unavoidable issue when dealing with most real-world data sources. Many researchers in the machine learning field have discussed and analyzed. Several important factors must be considered when dealing with unknown features.



- Multiply the new weights by the old weights of the model used for training the features together.
- The new weights are multiplied by the experimental precision of each separable sample before the voids are removed.

The dataset D can be split into two parts to define the missing data mathematically.

$$D = D_0, D_x$$

Here, D_0- detected data, D_x to the missing data in the Dataset.

$$F = \begin{cases} 1\ is\ \ D\ data\ detected \\ 0\ is\ D\ data\ missing \end{cases}$$

The missing data can be understood as values of possibility that a missing $Possibility\ (D)$ given the missing terms.

$$Possiblity(D|D_0, D_x)$$

This mechanism determines whether the possibility of Response ® depends on observed and missing values.

❖ **Data Normalization**

The Gaussian mixture model's possibility density function is a convex linear combination of the individual

Gaussian possibilities. It can estimate density for some Gaussian models in machine learning applications that use Gaussian mixture models. This is the product likelihood model. For each class label, compute the probability distribution. The oversampling ratio from the minority class distribution is then determined for each minority class event using the following formula:

$Min - \max\ normalization$:

$$M' = \frac{M - minimum_\lambda}{maxi_\lambda - min_\lambda}(new\_maxi_\lambda \ - new\_min_\lambda) + new\_min_\lambda$$

$$Z_{score}\ Normalization: M' = \frac{m - mean_\lambda}{std\_dev_\lambda}$$

m- The old feature values and M' the new features. Normalization is a "scale down" transformation of features. The features often have significant differences between their maximum and minimum values.

- ❖ **Training 70% and testing 30%**

The training and testing stages are separated to evaluate the performance of machine learning algorithms when they are used to make predictions on data that has yet to be used to develop a model.

A training dataset is used to fit a machine-learning model.

Test dataset: A dataset used to assess the fit of a machine learning model.

The scikit-learn Python library executes the train-test split assessment process with the train_test_split () capability.

3.3 Selecting sub-set features used Enhanced Binary Cuckoo Search with Ant colony optimization Algorithm (EBCS-ACO)

The proposed feature selection is an Enhanced Binary Cuckoo Search with Ant colony optimization Algorithm (EBCS-ACO). It uses both the feature selection methods efficiently for Use ACO (Ant Colony Optimization) to classify data and provide optimization parameters extracting maximum weighted features. However, Binary Cuckoo Search is an optimization technique that provides simple and easy parameter optimization.

Feature selection problem identification for more optimization definition, In the given feature Set $F = \{f_1 : s = 1, \dots, N\}$

Find a subset features $(f_s = \{f_{s_1}, f_{s_2,} \dots f_{s_n}\} with\ N < S)$

$$\left(f_{s_1}, f_{s_2,} \dots f_{s_n}\right) = \max_{N,s_1} argu[X\{f_1 : s = 1, \dots, N\}]$$

Selecting the optimal subset of features from the initial set involves a search strategy to pinpoint candidate subsets. EBCS-ACO analyses the feature weights of each feature. EBCS evaluates and selects the feature values based on its nearest feature threshold weights or values, and ACO estimates the sequencing of the order of the multiclass Dataset.

3.3.1 Enhanced Binary Cuckoo Search performance

Cuckoo parasitism is an exciting phenomenon. These birds lay eggs in the host's nest and replicate the host's outward traits, such as colouring and mottle. If this method fails, the host cuckoo may reject the eggs, quit the nest, and start over elsewhere. Using three rules, the enhanced Binary Cuckoo Search model summarises EBCS:

- Each cuckoo will haphazardly pick a home in which to lay its eggs.
- The number of accessible host homes is steady, and homes that lay excellent eggs are given to the future.
- If a host bird finds a cuckoo egg, it might either dispose of the egg or forsake the home and fabricate another home.

For enhancement issues, each home addresses a potential answer for the problem and may contain at least one egg, depending upon the extent of the case. The calculation then iteratively initializes each home aimlessly. Table 2 includes the section search boundaries.

Table 2: Binary Cuckoo Search Parameters

Parameters	Value	Description
α	0.1	Transition probability coefficient
β	1	Transition separation coefficient
ν	3%	Coefficient for the perturbation operator
N	20	Number of particles
G	8	Number of transition groups
Iteration	500	Maximum iteration

At each iteration, all nests are updated with a levy flight, and the formula is given by Eq.

$C_x^{s+1} = C_x^s + \alpha\Phi le'vy(\lambda)$

The updating the formula of each dimension is expressed as $C_{xy}^{s+1} = C_{xy}^x + \alpha X le'vy_y(\lambda)$

Where,$C_x^s - xth\ nest$,C_{xy}^{s+1}- $yth\ nest$ eggs at the nest $x\ for\ the\ s\ generation.\ \alpha$- is the step size, and the Φ means enterwise multiple data. $\alpha = 1$, Levy flight (λ) employs a random step length, and Levy flight (λ) is features selection.

Levy~n $= c^{-\lambda}(1 < \lambda \leq 3)$

$S = \frac{n}{|u|^{1/\beta}}$

S is the Levy flight (λ) relative parameters, β is estimate$\lambda = 1 + \beta, \begin{cases} \mu \sim S(0, \sigma_\mu^2) \\ \tau \sim S(0, \sigma_\tau^2) \end{cases}$,

$\left\{ \sigma_\mu = \left\{ \frac{\Gamma(1+\beta).\sin(\beta\Pi/2)}{\Gamma\left(1+\frac{\beta}{2}\right).\beta.2^{(\beta-1)/2}} \right\} \sigma_{\tau=1} \right.$

Let $\alpha \times$ Levy flight (λ) =$\alpha \times x$ then-step is the path a cuckoo takes to randomly search for a new nest location each time it resolves. C_{xy}^{s+1} From the old nest location C_x^s at each iteration, the worst nest is replaced with the Binary Cuckoo Search.

Steps for selecting feature for binary cuckoo search

Input: Total sample (S), class features (F), max_features, initial_features set ()

Output: Final Sub − selected features set (S)

Initialize selected features = null

For all features f_e in f do

Measure R_{info}-related information

Features set $F_{info} \cup x = f_e$

Set $F_{info} \cup x = 0$

End

For $N = 1\ to$ max_$features$, do

Feature $score_n = 0$

For every feature $F_{info}\ in\ f$ do

While $F_{info} > score_n\ AND\ f_e < n - 1\ do$

Features set $F \cup x = F \cup x + 1$

Computes CRI $_f$ between $and\ f_x$

$F_{info} = \min(f_e; Ef_{in})$

End while

If $Set\_score_n = f_x$

$F_{selectedfeature} = F_{selected} \cup \{f_x\}$

End

End

End

$Set\ (D) =\ final\ search\ selected\ features$

Receive feature scores and define CRIs (conditional relevance information) for informative and less redundant features during selection. Iteratively, the best N features are chosen to maximize the mutual information between them and the target class.

After EBCS takes N features, the set features are provided as input datasets to the EBCS algorithm, which selects the best-suited subset of features to increase classification accuracy and perform sophisticated multiplexing. Predict the feature with the highest maximum weight from a single dataset.

In classical CS, the solution spaces in a sequential search space are updated. In contrast to the preceding CS, the EBCS scan space for highlight choice is addressed by an extended piece string mirroring the number of elements. EBCS represents each joint as an enhancement of the feature maximum weight vector. One for each feature selected, 0 otherwise. Each nest is an activity, and each nest represents a possible solution.

$f_{xy} = \begin{cases} 1\ f_x\ select\ the\ yth\ feature \\ 0\ otherwise \end{cases}$

The first Cuckoo calculation presents a planning capability, which stretches out the Cuckoo calculation to the weighted space as follows:

$sig(stage) = \frac{1}{1+c^{-\gamma stage}}, \gamma = 1$ (9)

$f_{xy}^s = \left\{ \begin{cases} 1\ random\ () \leq sig(stage) \\ 0\ otherwise \end{cases} \right.,$

In which random () ~∪ (0,1) and f_{xy}^s Iteration (i) returns the new egg value. Binary Cuckoo Search explores the search space in a linear path with random selections using a Lévy plane search approach. Furthermore, CS is mainly based on random walk searches, which allow it to quickly jump from one location to another without thoroughly inspecting each cuckoo's nest.

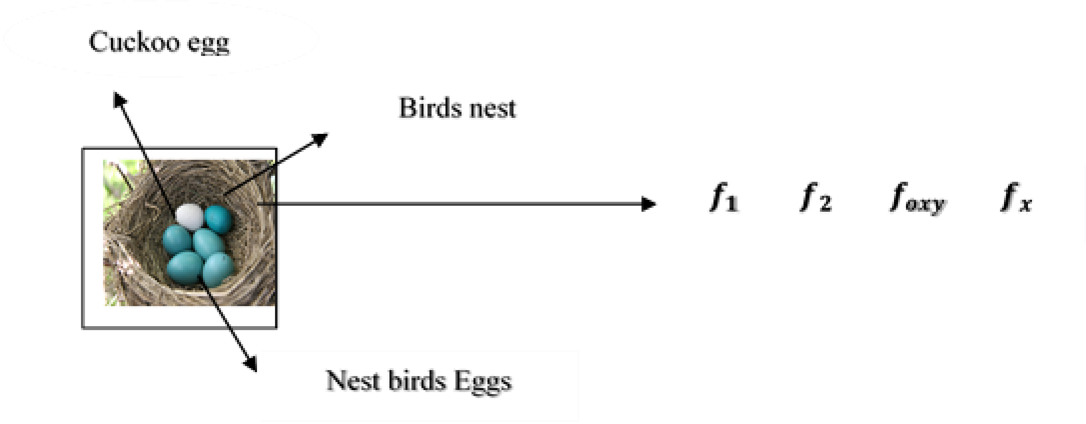


Figure 3: Enhanced Binary Cuckoo Search Algorithm Representation

- Figure 3 depicts a nest in which all existing or available eggs contribute solutions, and cuckoo eggs provide new ones.
- This strategy is used to enhance the worst Nest solutions. Said every nest has an egg in it.
- The method becomes more complicated when each nest contains numerous eggs, indicating a set of solutions.

Thus, CS has the following disadvantages: poor optimization performance, slow convergence speed, weak local search. To overcome these shortcomings, the improved Binary Cuckoo Search algorithm shows better selection performance.

✓ **Weight-Based Enhanced Binary Cuckoo Search**

A levy plane is used in the EBCS calculation to cross the hunt space using a straight flight course with sharp 90-degree revolutions. Figure 3 portrays the levee's flight course. Moreover, the CS calculation chiefly depends on an irregularity, which permits it to effortlessly head out, starting with one region and then onto the next without thoroughly examining each home. To address CS's shortcomings, search procedure drivers, for example, choice and crossing point administrators, are coordinated into the Cuckoo calculation, permitting all-around set homes to be given in the future. The supposed pursuit procedure is to store the houses in the right area so that the best homes are not missed through the Lévy plane calculation cycles. A choice administrator follows explicit methods to guarantee that fit people from the ongoing populace are passed down to the future.

Steps for maximum fitness selection based on feature weights

Input: Population with n nests, number of features (f), fitness function f(u)

Output: new features in new nests after selection

For each x (a = 1, …, n) do

$n(f_x)/\sum_{y=1}^{n} f(f_z)$

$m_x = \sum_{y=1}^{n} n(f_z)$

For each z (z = 1,…, n) do

Select a random number n_j from [0, 1];

If ($n_j \leq m_x$) then

Select the n_x;

Else

If ($n_{j-1} < f_x \leq m_x (2 \leq x \leq n)$) then

Select the f_x;

End

End

End

End

Where $n(f_x)$ is the selection possibility, n_j is the increasing probability, m_x is the individual $f$itness function value, and n is the group number.

3.3.2 Ant colony optimization (ACO) performance

A Feature Selection Approach Based on Ant Colony Optimisation Algorithm (ACO) Integration to Improve Text Classifier Performance. ACO is a heuristic search algorithm inspired by research on real ant foraging behaviour, namely pheromone transmission, to locate the shortest path to food sources.

ACO is a process consisting of three main steps. The ants generate multiple solutions at each iteration. Local searches will improve the solution; eventually, pheromones will be updated based on their quality.

$$A_{xy}(s) = \begin{cases} \frac{[\mu_{xy}(s)]^{\alpha}[\eta_{xy}(s)]^{\beta}}{\sum_{local}[\mu_{xy}(s)]^{\alpha}[\eta_{xy}(s)]^{\beta}} & if\ y\epsilon \text{ Allowed} \end{cases}$$

Where α and β = parameters that control the relative importance of the pheromone pathway and the heuristic value. Let q be [0, 1], and n0 ∈ [0, 1] be the only random variable in the parameter. Ant (n) selects node j as:

$$n = \begin{cases} \arg\text{maximum}_{x\epsilon allowed}[\mu_{xy}(s)]^{\alpha}[\eta_{xy}(s)]^{\beta} \\ N\ otherwise \end{cases}$$

Where n = x is a random variable chosen according to a probability distribution $\mu_{xy}(s)$, the pheromone trail is changed globally.

$$\Delta\mu_{xy}(s) = \sum_{n=1}^{n} \mu n_{xy}^{x}\ (s)$$

$$\mu n_{xy}^{x}(s) = \begin{cases} \frac{1}{A^{n}(x)} & if\ (x,y) \in A^{t}(x) \\ 0 & if\ \notin A^{t}(x) \end{cases}$$

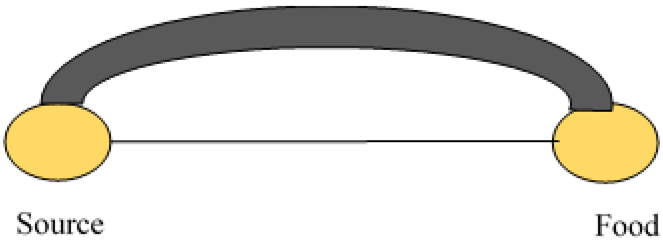
$A^{t}(x)$, it is Taken by the iteration function (x). Find ACO system iteration Best (ACIB)

$$\Delta\mu_{xy}(s) = \begin{cases} A^{n} & if\ (x,y) done \quad (x) \\ 0 & if\ \notin A^{t}(x) \end{cases}$$

Where A^{n} The value of the objective function of ant is the best performance within the last total iterations.

The operating system of subterranean insects to find their food with the assistance of the briefest way is exceptionally engaging.

- Every ant can remember their path.
- These mechanisms evaporate with time.



Primary State of finding food from source for ANT

Behaviour of ANTs

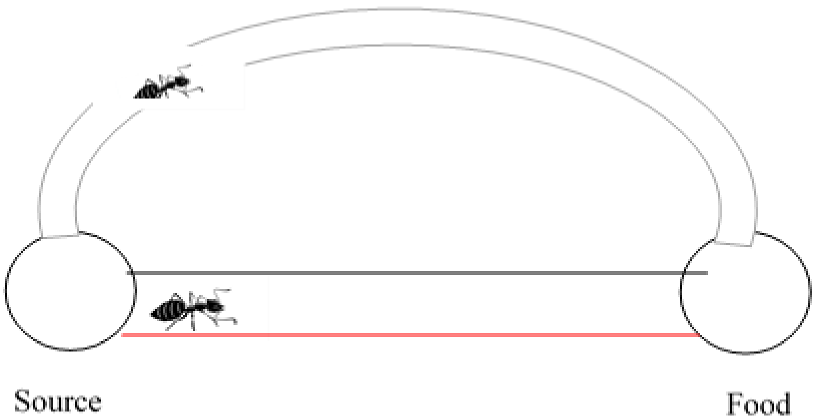


Figure 4 (a) Ants start with an equal probability of going on either path.

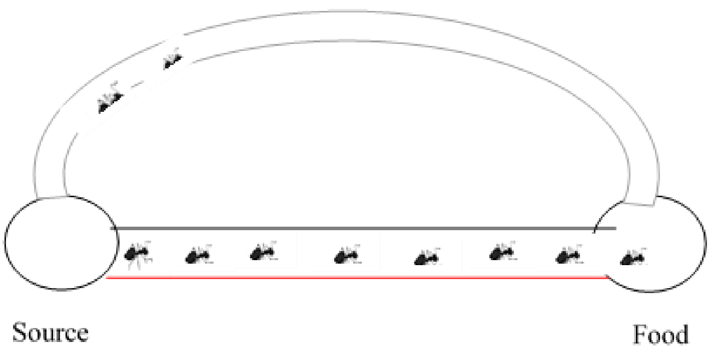


Figure 4 (b) More ants begin using the path with a higher efficient sequence order.

Figure 4 (a) and (b) depict the technique of using pheromones to record the return path of ants after they find food.

- Ant trails aid in determining the shortest path.
- All other ants take the shortest path to food to survive.

➢ All the ants compete to find the quickest path discovered by their companions.

➢ Finally, the ants take the shortest route to the food. Table 3 explains the ant parameters using the following functions.

Table 3: Basic Parameters of ACO

Parameter Name	Meaning	Initial values
N	No.of.Ants	20
g	No.of.Generation	50
$\tau 0$	initialisation of pheromone	0.5
α	weight of Pheromone on decision	1
β	The influence of heuristic data on decision-making	0.2
Q	amount of Pheromone to be deposited	2
ρ	percentage of pheromone evaporation during one step	0.05

3.3.3 EBCS with ACO Feature Selection

We present a half-breed choice methodology that considers the general nature of the produced, including the subset and the singular pertinence of the elements. The regular ACO calculation displayed in Calculation 1 is utilized in this work. The EBCS-ACO highlights choice calculation. Using arrangement procedures, assess the presentation of element subsets. The importance of a given viewpoint then turns into a weight estimated utilizing EBCS.

$$F_i^n = \begin{cases} \frac{\mu_i^\alpha \beta_i^\alpha}{\sum_{notallow} \mu_i^\alpha \beta_i^\alpha} \end{cases}$$

After initialization, the cloak generates various possible subsets. Each ant k randomly chooses an initial feature to form its feature subset F_i^n; each ant autonomously selects the following selected characteristic until a complete subset is produced.

The α and β factors determine the relative importance of pheromones to heuristic information. And $\sum_{notallow} \mu_i^\alpha \beta_i^\alpha$ It is the sum of products $\tilde{\mu}_i^\alpha \beta_i^\alpha$ All features ant x partial solution.

Step EBCS-ACO

Input: data features samples for training $f_1, f_2, .. f_n$

Output: Best feature for Classification

subset Initialize the EBCS Weight parameters.

➢ Create a search query
➢ Start ACO parameters
➢ Pheromone initiation;
➢ The number of ants;

For each ant A, do

Repeat Choose in probability the feature to include;

Use EBCS scores to adjust possibility selection;

Append the partial solution with the candidate feature;

Until the ant (A) chooses a sequence feature

- Assess the developed division$-f_x$;
- Using classifier accuracy to evaluate
- The candidates include a subset;

If (end condition not met) do

For each featuref_x; used in c_x

Update Pheromone in light of the performance quality

End for

The accuracy of the classifier determines the quality of the generated solution in this work. c_x, and the quantity of the Pheromone arranged on featuref_x is

$$\Delta\mu_x = \sum_{x=1}^{n} \Delta\mu_x^n$$

Where m is the number of ants and $\Delta\mu_x^n$, the level of Pheromone is set for the feature μ_x. AntN scored each feature, and the weighted part with the highest score was selected for element reduction based on EBCS-ACO.

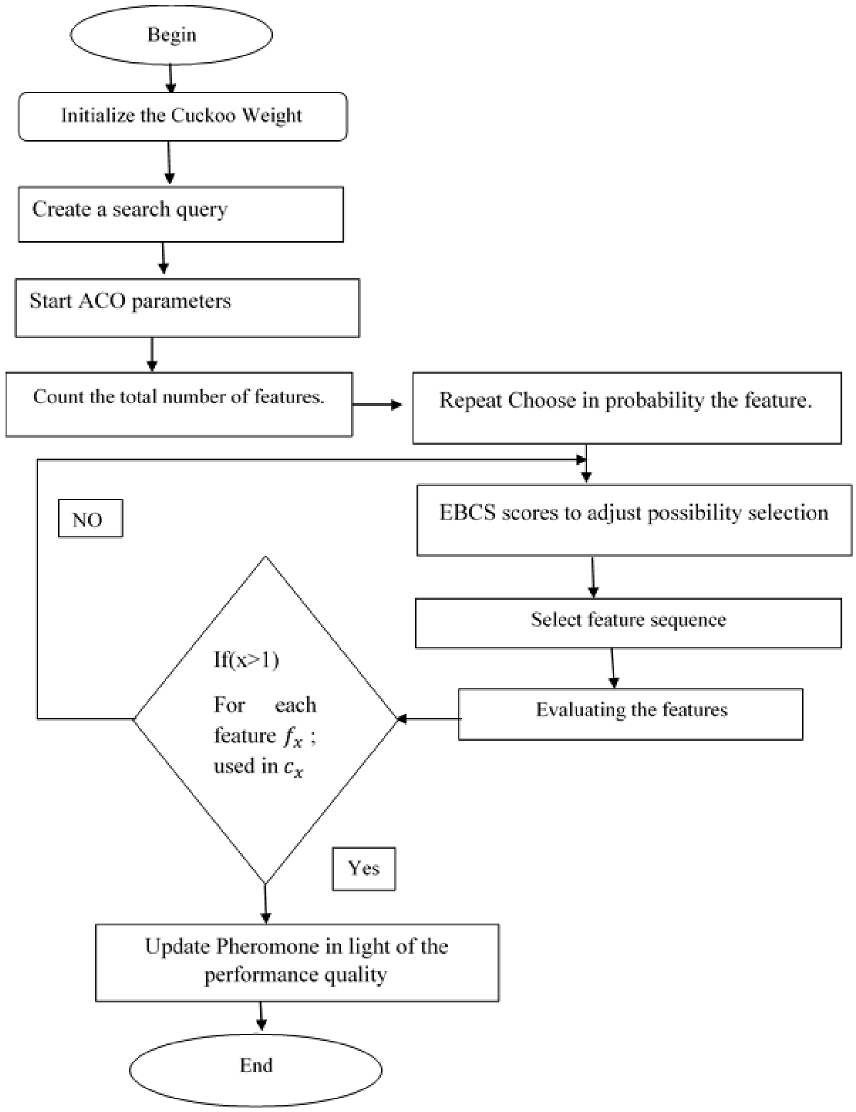


Figure 5: Flow chart for EBCS with ACO

Figure 5 Flow chart for EBCS Combined with ACO, it collects the feature weights to evaluate the ACO and the sequence; all features are used to update the performance; otherwise, it checks the score of EBCS.

3.4 K-fold cross-validation

The available preparation set has been partitioned into around 70,000 disjoint subsets of equivalent size in k-fold cross-validation.

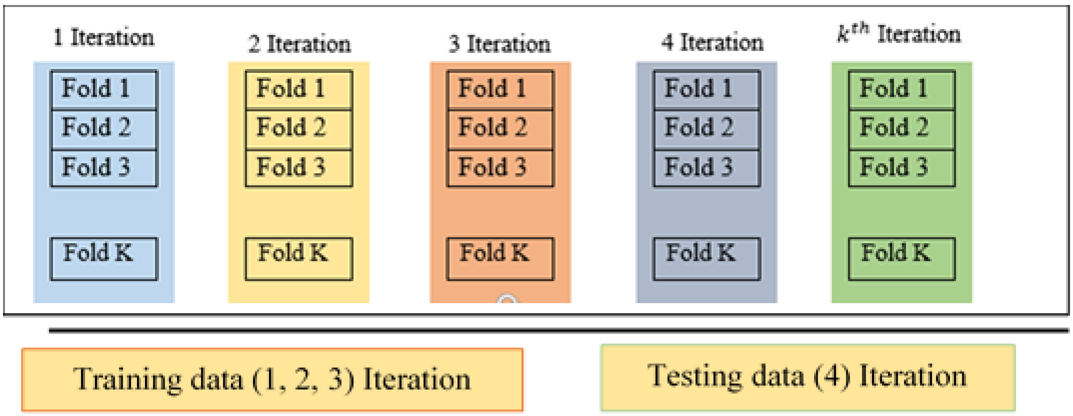


Figure 6: K-fold Validation

Steps for K-fold cross-validation

Stage 1→ separating the training data (Df_{train}) into K equalizes the subset features such as $(f_1, f_2, .. f_n)$

Stage 2→For $f = 1\ to\ x = k$

Stage 3→ $\mathrm{k} - 1$ folds the training dataset and y as the testing dataset

Stage 4→ machine learning model (m) and compute accuracy (A)

Stage 5→ K instance of accuracy to evaluate

The term "fold" refers to the number of subgroups that result. This segmentation is accomplished by randomly picking events from the training set and leaving them alone. k - 1 subsets are utilized to prepare a model. The preparation set contains these subgroups. The model is then applied to the excess subset, alluded to as the approval set, and its presentation is assessed in Figure 6.

3.5 Classification using KNN

K-Nearest Neighbors (KNN) is an essential and typically effective non-parametric supervised classification technique. KNN classifiers are the most used pattern recognition because of their efficient performance, results, and simplicity.

It is implicitly believed that the K nearest neighbors will weight equally regardless of their distance from the experimental data. The K nearest neighbors should be assigned varying weights based on how close or far they are from the experimental data, with the more immediate neighbors receiving more weight.

$$K = \max_{f \in dis(K)} arg \sum_{f_{x \in KNN(S)}} w(k_x == F) \frac{1}{f(f_x, f)}$$

Where $F(f_x, f)$ is the distance (f_x, f) and the best data features $(f).Max_{f_{x \in KNN(S)}}\{(f_x, f)\}$.

$$K = \max_{f \in dis(K)} arg \sum_{f_{x \in KNN(S)}} w(k_x == F)(1 - \frac{1}{f(max)})$$

Weighting classification:

$$K = \max_{f \in dis(K)} arg \sum_{f_{x \in KNN(S)}} wx * 1(k_x == F)$$

A kernel function classification

$$K = \sum_{f_{x \in KNN(S)}} wxK((f_x, f)$$

Where $K(f_x, f)$ or a numeric value. If the determining attribute is a specific character type, Yi can be replaced by a pointer function.

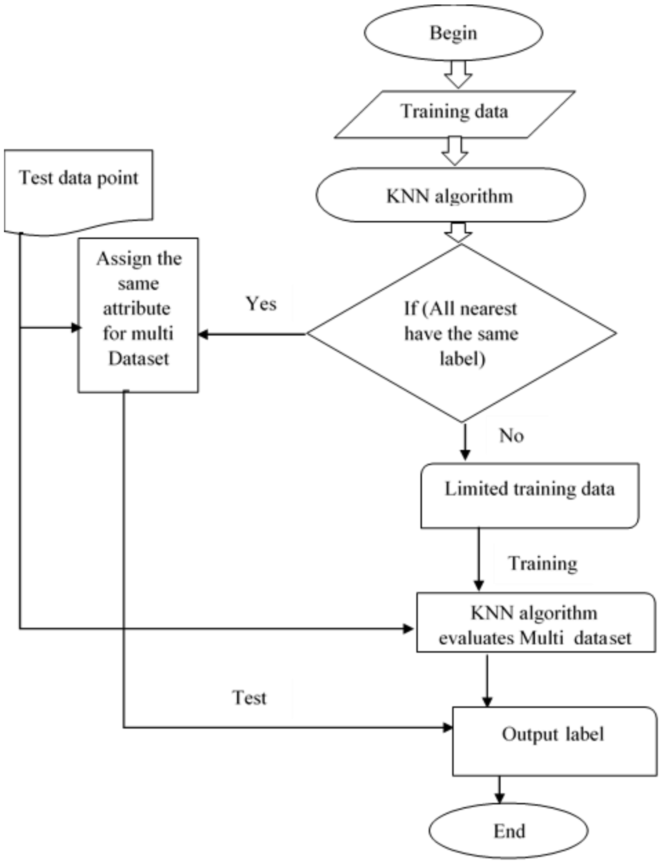


Figure 7: Flowchart for KNN classification algorithm

Figure 7 shows a new, unlabelled test point xs first, then locates nearby K training points in the feature space. The KNN model is trained using only the new subset based on the nearest K points, creating a different model for each test point. The benefit of this local strategy is that it improves classification performance when there is class disparity and is computationally and time economical, especially for online modelling and flow analysis.

4. Result and analysis

Simulation results providing Precision, recall, and F-Measure are three performance analysis indicators used to evaluate the proposed system's performance. The proposed system's performance is compared to an existing system that classifies multiclass data features using a KNN classifier.

Table 4: Simulation parameters

Parameters	Values
Collection of Dataset names	Glass -214 samples, Letter -20000 samples, Vehicle -846 samples, Spambase-4601 samples, Waveform - 5000 samples.
Tool	Anaconda
Language	Python
Training Records	70%
Testing Records	30%

Table 4 describes the multiclass dataset features evaluation using the KNN classifier for better accuracy than existing methods. It concentrates the selection based on the maximum weighted features and sequencing order selection. It collects the training and testing records classified using the Python language and Anaconda tool.

Table 5: Analysis of Precision and Recall

Methods/Dataset	MOGWO	BACOHAFSS	Proposed	MOGWO	BACOHAFSS	Proposed
Glass	60	68	75	60	64	72
Letter	70	72	86	68	70	83
Spambase	78	80	88	70	75	87
Vehicle	80	85	89	79	83	88
Waveform	81	88	92	80	84	91

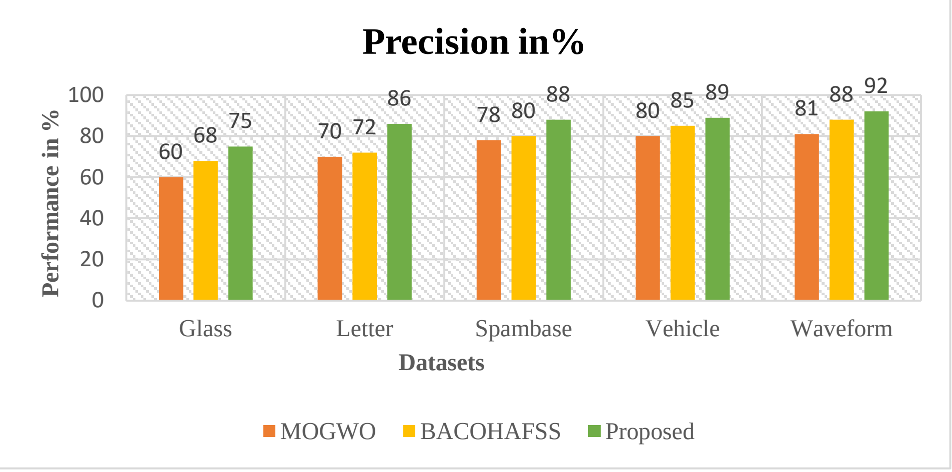


Figure 8 (a) Precision Analysis

Table 5 and Figure 8 (a) defines the precision level of different dataset record findings in the functional evaluation to classify the results. For the proposed approval, the multiclass feature dataset produces 92% precision for testing features and 88% for training features.

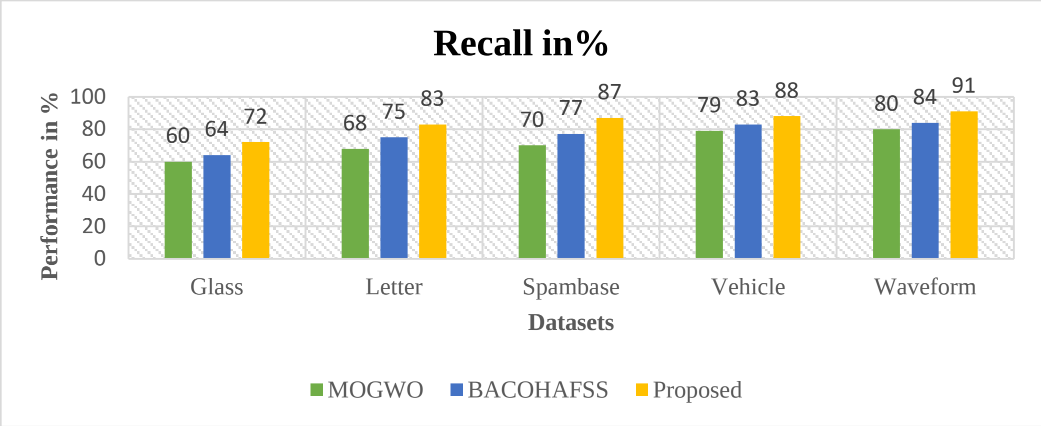


Figure 8(b) Analysis of the recall performance

Table 5 and Figure 8(b) shows the so-called actual values produced in various ways, where the proposed method outperforms the different. For the suggested esteem, the multiclass feature dataset yields a testing feature recall of 91% and a training feature recall of 84%.

Table 6: Analysis of Accuracy and Error Rate

Dataset	MOGWO (Accuracy)	BACOHAFSS (Accuracy)	Proposed (Accuracy)	MOGWO (Error Rate)	BACOHAFSS (Error Rate)	Proposed (Error Rate)
Glass	0.82	0.857	0.894	0.18	0.143	0.106
Letter	0.86	0.926	0.951	0.14	0.074	0.049
Spam base	0.88	0.942	0.962	0.12	0.058	0.038
Vehicle	0.69	0.798	0.868	0.31	0.202	0.132
Waveform	0.91	0.935	0.961	0.09	0.065	0.039

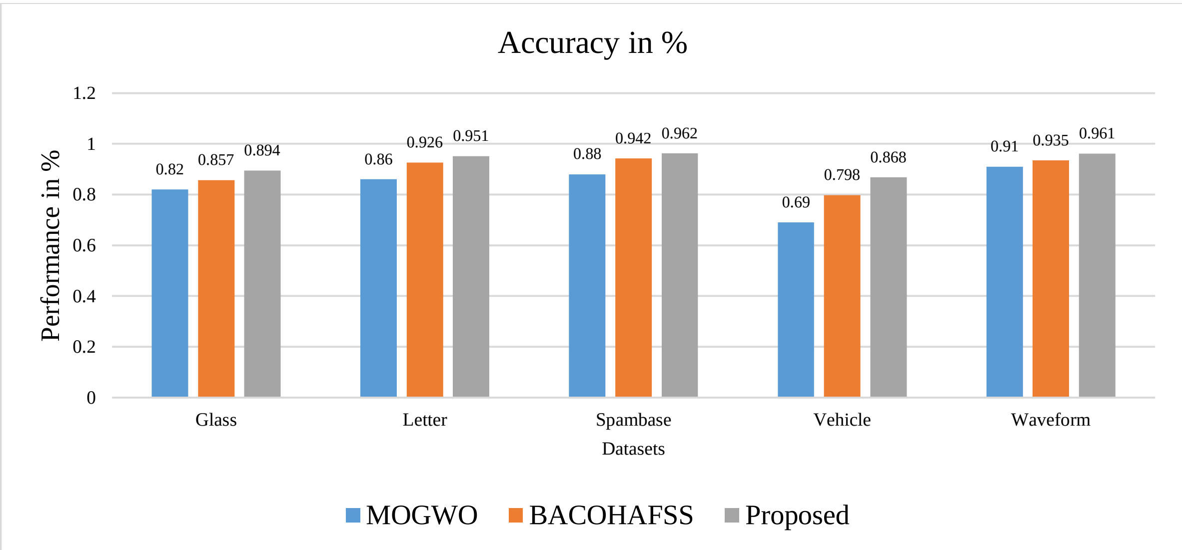


Figure 9(A) classification Accuracy

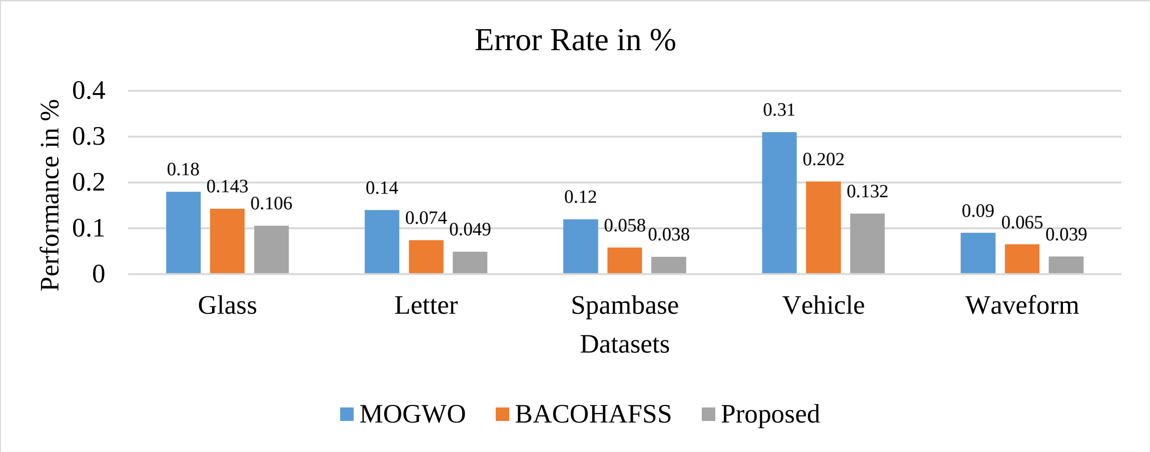


Figure 9 (B) Analysis of Error Rate

As absolute results are displayed based on type class, Classification establishes the accuracy and error rate uniqueness of frequent measurements anticipated by fit/accuracy provided by positive values. Table 6 and Figure 9 (A) ,9 (B) display the Classification's accuracy and error rate better than the previous method.

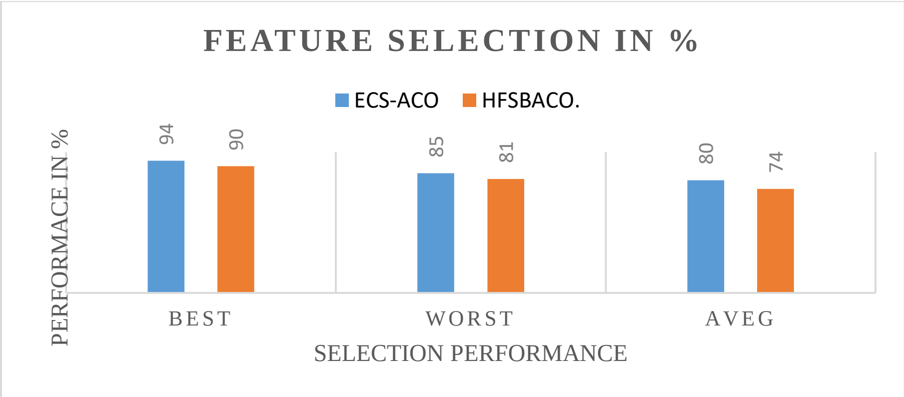


Figure 10 Feature selection performance

Figure 10 and table 5 shows the selection of the experimental findings, demonstrating that, for all datasets, the typical element subset expands to shifting degrees of the usual grouping precision. The average number of feature subsets after feature selection is more similar to that of the feature selection process than it is to the original Dataset that has been applied in various Datasets such as Glass -214 samples, Letter -20000 samples, Vehicle -846 samples, Spambase-4601 samples, Waveform - 5000 samples. Based on ML optimization algorithms, these feature selection techniques can significantly increase classification accuracy by successfully removing superfluous information.

Table 7: F-measure analysis

Dataset	MOGWO	BACOHAFSS	Proposed
Glass	55	60	65
Letter	60	65	70
Spambase	65	70	74
Vehicle	68	75	79
Waveform	73	80	84

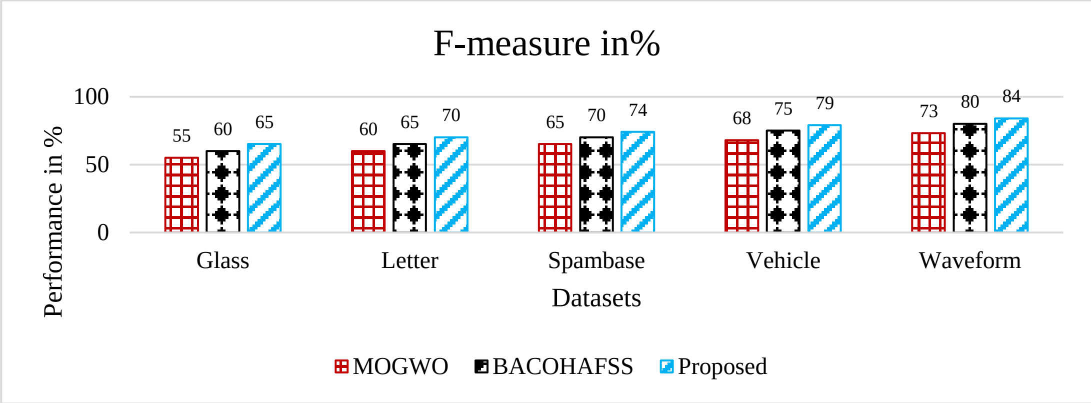


Figure 11: F-measure Evaluation

Precision and recall are given equal weight when calculating the F-measure. This makes it possible to evaluate the model using single-score feature selection for Precision and recall, which helps assess performance in Figure 11 and table 5.

Table 8: Time complexity

Dataset	MOGWO	BACOHAFSS	Proposed
Glass	11.8	9.3	8.2
Letter	10.9	9	7.9
Spambase	9.6	8.9	8.4
Vehicle	9.8	8.5	8.2
Waveform	9.3	8.3	6.3

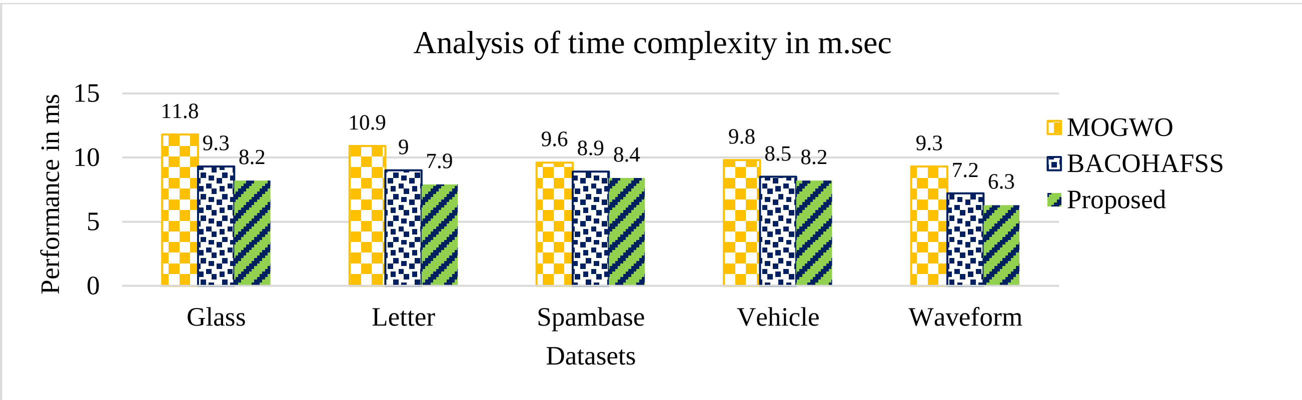


Figure 12: Impact of time complexity

Table 8 and figure 12 shows the detection accuracy is based on the processing time. Compare the detection accuracy of different methods. O (n) will process all the records based on the type definition of the defined type so that the best detection will take more time. As shown in Figure 12, the proposed system takes 6.3 (ms) less time for the feature selection performance than all previous systems.

Table 9: Proposed simulation result

Dataset	Classifier	Accuracy (%)	F-measure (%)
Glass	KNN	0.894	65
Letter	KNN	0.951	70
Spambase	KNN	0.962	74
Vehicle	KNN	0.868	79
Waveform	KNN	0.961	84

Table 7 illustrates the proposed simulation accuracy and F-measure performance for multiclass feature selection and Classification.

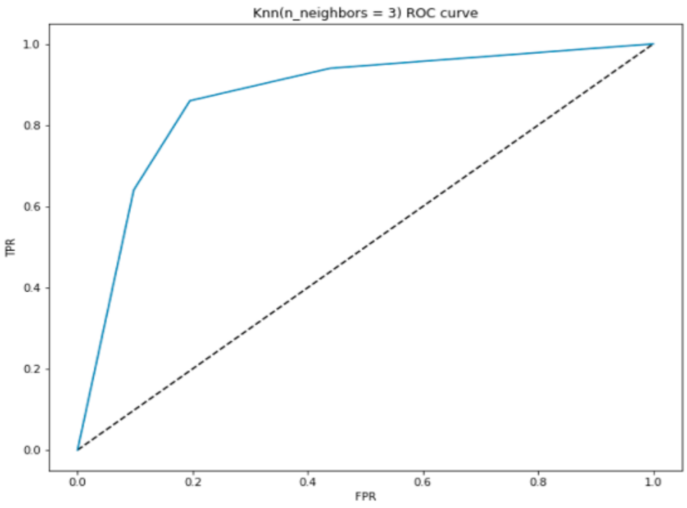


Figure 13: ROC curve analysis for multiclass Classification

Figure 13 describes the ROC curve analysis of multiclass Classification. ROC curves are utilized to assess the performance of classification models and illustrate the relationship between True Positive Rate (TPR) and False Positive Rate (FPR). Here, n_neighbors =3 It describes the nearest class of multi Dataset.

5. Conclusion

Multiclass feature selection (FS) is a prevalent technique in machine learning, mainly when working with high-dimensional datasets. By eliminating redundant Highlights with practically no prescient worth or solid relationship, the essential objective of component determination is to pick a subset of the accessible elements. The accessibility of a great deal of information makes grouping examinations difficult. For instance, while dealing with many features, estimating a sizable number of parameters may be essential. Ideally, each element in the classification process should have a separate piece of information added to it. We suggest a KNN using feature weighting and search techniques and EBCS-ACO. The proposed KNN technique aims to enhance binary label dataset feature selection jobs. A drawback of this technique is that multiclass selection functions have a lower computational complexity. The proposed methodology should be further modified in subsequent work to assess the results using other datasets and classification models that can accommodate feature selection in multiclass datasets.

Reference

[1] Ruchika Singh Rajput, Dr. Jitendra Agrawal, Dr. Sanjeev Sharma"Binary Cuckoo Search based Hybrid Classification Techniques", IJCST Vol. 8, Issue 1, Jan - March 2017.

[2] R.Senthamil Selvi, K.Fathima Bibi, “A Machine Learning-Based Hybrid Approach to Subset Selection Using Binary Ant Colony Optimization Functions”, SN Computer Science vol.4,no.853,pp. 1-7,8 November 2023,doi: https://doi.org/10.1007/s42979-023-02251-9.

[3] Q. Lou, Z. Deng, K. -S. Choi, H. Shen, J. Wang and S. Wang, "Robust Multilabel Relief Feature Selection Based on Fuzzy Margin Co-Optimization," in IEEE Transactions on Emerging Topics in Computational Intelligence, vol. 6, no. 2, pp. 387-398, April 2022, doi: 10.1109/TETCI.2020.3044679.

[4] K. Yu, L. Liu, J. Li, W. Ding and T. D. Le, "Multi-Source Causal Feature Selection," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 42, no. 9, pp. 2240-2256, 1 Sept. 2020, doi: 10.1109/TPAMI.2019.2908373.

[5] Z. Xiong, Y. Yuan and Q. Wang, "RGB-D Scene Recognition via Spatial-Related Multimodal Feature Learning," in IEEE Access, vol. 7, pp. 106739-106747, 2019, doi: 10.1109/ACCESS.2019.2932080.

[6] M. Usman, U. K. Yusof and S. Naim, "Filter-Based Multiobjective Feature Selection Using NSGA III and Cuckoo Optimization Algorithm," in IEEE Access, vol. 8, pp. 76333-76356, 2020, doi: 10.1109/ACCESS.2020.2987057.

[7] Y. Zhang, D. -w. Gong and J. Cheng, "Multiobjective Particle Swarm Optimization Approach for Cost-Based Feature Selection in Classification," in IEEE/ACM Transactions on Computational Biology and Bioinformatics, vol. 14, no. 1, pp. 64-75, 1 Jan.-Feb. 2017, doi: 10.1109/TCBB.2015.2476796.

[8] T. Xu and L. Zhao, "A Structure-Induced Framework for Multilabel Feature Selection With Highly Incomplete Labels," in IEEE Access, vol. 8, pp. 71219-71230, 2020, doi: 10.1109/ACCESS.2020.2987922.

[9] X. Zhu, S. Zhang, Y. Zhu, P. Zhu and Y. Gao, "Unsupervised Spectral Feature Selection With Dynamic Hyper-Graph Learning," in IEEE Transactions on Knowledge and Data Engineering, vol. 34, no. 6, pp. 3016-3028, 1 June 2022, doi: 10.1109/TKDE.2020.3017250.

[10] D. R. Wijaya and F. Afianti, "Information-Theoretic Ensemble Feature Selection With Multi-Stage Aggregation for Sensor Array Optimization," in IEEE Sensors Journal, vol. 21, no. 1, pp. 476-489, 1 Jan.1, 2021, doi: 10.1109/JSEN.2020.3000756.

[11] L. Y. Yab, N. Wahid and R. A. Hamid, "A Meta-Analysis Survey on the Usage of Meta-Heuristic Algorithms for Feature Selection on High-Dimensional Datasets," in IEEE Access, vol. 10, pp. 122832-122856, 2022, doi: 10.1109/ACCESS.2022.3221194.

[12] N. L. S. Albashah and H. M. Rais, "Population Initialization Factor in Binary Multiobjective Grey Wolf Optimization for Features Selection," in IEEE Access, vol. 10, pp. 114942-114958, 2022, doi: 10.1109/ACCESS.2022.3218056.

[13] Q. Al-Tashi et al., "Binary Multiobjective Grey Wolf Optimizer for Feature Selection in Classification," in IEEE Access, vol. 8, pp. 106247-106263, 2020, doi: 10.1109/ACCESS.2020.3000040.

[14] G. Sharifai and Z. B. Zainol, "Multiple Filter-Based Rankers to Guide Hybrid Grasshopper Optimization Algorithm and Simulated Annealing for Feature Selection With High Dimensional Multiclass Imbalanced Datasets," in IEEE Access, vol. 9, pp. 74127-74142, 2021, doi: 10.1109/ACCESS.2021.3081366.

[15] M. Ramona, G. Richard and B. David, "Multiclass Feature Selection With Kernel Gram-Matrix-Based Criteria," in IEEE Transactions on Neural Networks and Learning Systems, vol. 23, no. 10, pp. 1611-1623, Oct. 2012, doi: 10.1109/TNNLS.2012.2201748.

[16] S. D. A. Bujang et al., "Multiclass Prediction Model for Student Grade Prediction Using Machine Learning," in IEEE Access, vol. 9, pp. 95608-95621, 2021, doi: 10.1109/ACCESS.2021.3093563.

[17] J. Wu, P. Guo, Y. Cheng, H. Zhu, X. -B. Wang and X. Shao, "Ensemble Generalized Multiclass Support-Vector-Machine-Based Health Evaluation of Complex Degradation Systems," in IEEE/ASME Transactions on Mechatronics, vol. 25, no. 5, pp. 2230-2240, Oct. 2020, doi: 10.1109/TMECH.2020.3009449.

[18] M. K. Keleş and Ü. Kılıç, "Artificial Bee Colony Algorithm for Feature Selection on SCADI Dataset," 2018 3rd International Conference on Computer Science and Engineering (UBMK), Sarajevo, Bosnia and Herzegovina, 2018, pp. 463-466, doi: 10.1109/UBMK.2018.8566287.

[19] Z. Wang, X. Xiao and S. Rajasekaran, "Novel and efficient randomized algorithms for feature selection," in Big Data Mining and Analytics, vol. 3, no. 3, pp. 208-224, Sept. 2020, doi: 10.26599/BDMA.2020.9020005.

[20] L. Gong, S. Xie, Y. Zhang, M. Wang and X. Wang, "Hybrid Feature Selection Method Based on Feature Subset and Factor Analysis," in IEEE Access, vol. 10, pp. 120792-120803, 2022, doi: 10.1109/ACCESS.2022.3222812.

[21] S. Li and D. Wei, "Extremely High-Dimensional Feature Selection via Feature Generating Samplings," in IEEE Transactions on Cybernetics, vol. 44, no. 6, pp. 737-747, June 2014, doi: 10.1109/TCYB.2013.2269765.

[22] C. Chen, Y. Wan, A. Ma, L. Zhang and Y. Zhong, "A Decomposition-Based Multiobjective Clonal Selection Algorithm for Hyperspectral Image Feature Selection," in IEEE Transactions on Geoscience and Remote Sensing, vol. 60, pp. 1-16, 2022, Art no. 5541516, doi: 10.1109/TGRS.2022.3216685.

[23] S. Wang et al., "Research and Experiment of Radar Signal Support Vector Clustering Sorting Based on Feature Extraction and Feature Selection," in IEEE Access, vol. 8, pp. 93322-93334, 2020, doi: 10.1109/ACCESS.2020.2993270.

[24] Q. Yu, J. Qian, S. Jiang, Z. Wu and G. Zhang, "An Empirical Study on the Effectiveness of Feature Selection for Cross-Project Defect Prediction," in IEEE Access, vol. 7, pp. 35710-35718, 2019, doi: 10.1109/ACCESS.2019.2895614.

[25] Spolaôr, Newton; Cherman, Everton Alvares; Monard, Maria Carolina; Lee, Huei Diana (2013). A Comparison of Multilabel Feature Selection Methods using the Problem Transformation Approach. Electronic Notes in Theoretical Computer Science, 292(), 135–151. doi:10.1016/j.entcs.2013.02.010.

[26] N. Laopracha, K. Sunat and S. Chiewchanwattana, "A Novel Feature Selection in Vehicle Detection Through the Selection of Dominant Patterns of Histograms of Oriented Gradients (DPHOG)," in IEEE Access, vol. 7, pp. 20894-20919, 2019, doi: 10.1109/ACCESS.2019.2893320.

[27] L. Sun, T. Yin, W. Ding and J. Xu, "Hybrid Multilabel Feature Selection Using BPSO and Neighborhood Rough Sets for Multilabel Neighborhood Decision Systems," in IEEE Access,

vol. 7, pp. 175793-175815, 2019, doi: 10.1109/ACCESS.2019.2957662.

[28] X. -T. Wang and X. -Z. Luan, "Bayesian Penalized Method for Streaming Feature Selection," in IEEE Access, vol. 7, pp. 103815-103822, 2019, doi: 10.1109/ACCESS.2019.2930346.

[29] F. Nie, S. Yang, R. Zhang and X. Li, "A General Framework for Auto-Weighted Feature Selection via Global Redundancy Minimization," in IEEE Transactions on Image Processing, vol. 28, no. 5, pp. 2428-2438, May 2019, doi: 10.1109/TIP.2018.2886761.

[30] Y. Tian, J. Zhang, L. Li and Z. Liu, "A Novel Sensor-Based Human Activity Recognition Method Based on Hybrid Feature Selection and Combinational Optimization," in IEEE Access, vol. 9, pp. 107235-107249, 2021, doi: 10.1109/ACCESS.2021.3100580.

[31] Wiharto, E. Suryani, S. Setyawan and B. P. Putra, "The Cost-Based Feature Selection Model for Coronary Heart Disease Diagnosis System Using Deep Neural Network," in IEEE Access, vol. 10, pp. 29687-29697, 2022, doi: 10.1109/ACCESS.2022.3158752.

[32] H. C. S. C. Lima, F. E. B. Otero, L. H. C. Merschmann and M. J. F. Souza, "A Novel Hybrid Feature Selection Algorithm for Hierarchical Classification," in IEEE Access, vol. 9, pp. 127278-127292, 2021, doi: 10.1109/ACCESS.2021.3112396.

[33] Malek Alzaqebah;Khaoula Briki;Nashat Alrefai;Sami Brini;Sana Jawarneh;Mutasem K. Alsmadi;Rami Mustafa A. Mohammad;Ibrahim ALmarashdeh;Fahad A. Alghamdi;Nahier Aldhafferi;Abdullah Alqahtani; (2021). *Memory based Binary Cuckoo Search algorithm for feature selection of gene expression dataset . Informatics in Medicine Unlocked, (),* – . doi:10.1016/j.imu.2021.100572

[34] R. Devi Priya, R. Sivaraj, N. Anitha, V. Devisurya, Tri-staged feature selection in multiclass heterogeneous datasets using memetic algorithm and Binary Cuckoo Search optimization,Expert Systems with Applications,Volume 209,2022,118286,ISSN 0957-4174,https://doi.org/10.1016/j.eswa.2022.118286.

[35] Kashef, Shima; Nezamabadi-pour, Hossein (2015). An advanced ACO algorithm for feature subset selection. Neurocomputing, 147(), 271–279. doi:10.1016/j.neucom.2014.06.067.

[36] Esra Saraç, Selma Ayşe Özel, "An Ant Colony Optimization Based Feature Selection for Web Page Classification", The Scientific World Journal, vol. 2014, Article ID 649260, pp. 1-14, 2014.

[37] Gite, S.; Patil, S.; Dharrao, D.; Yadav, M.; Basak, S.; Rajendran, A.; Kotecha, K. Textual Feature Extraction Using Ant Colony Optimization for Hate Speech Classification. Big Data Cogn. Comput. 2023, 7, 45.

[38] NK. Sreeja, A. Sankar,Pattern Matching based Classification using Ant Colony Optimization based Feature Selection, Applied Soft Computing, Volume 31, 2015, Pages 91-102.

[39] Chakravarty, Sujata & Mohapatra, Puspanjali. (2015). Multiclass Classification using Binary Cuckoo Search-based hybrid network. 953-960. 10.1109/PCITC.2015.7438134.

[40] R. Devi Priya, R. Sivaraj, N. Anitha, V. Devisurya,Tri-staged feature selection in multiclass heterogeneous datasets using memetic algorithm and Binary Cuckoo Search optimization,Expert Systems with Applications,Volume 209, 2022.

[41] Qinwei Fan, Tongke Fan, "A Hybrid Model of Extreme Learning Machine Based on Bat and Binary Cuckoo Search Algorithm for Regression and Multiclass Classification", Journal of Mathematics, vol. 2021, Article ID 4404088, pp. 11, 2021.