IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.DOI

Metric and Accuracy Ranked Feature Inclusion: Hybrids of Filter and Wrapper Feature Selection Approaches

THEJAS G.S.[1,2], (Member, IEEE), RAMESHWAR GARG[3], S.S. IYENGAR [2],(LIFE FELLOW, IEEE), N.R SUNITHA [4],(Member, WIE, IEEE), PRAJWAL BADRINATH[2], SHASANK CHENNUPATI[5]

[1]Department of Computer Science and Electrical Engineering, Tarleton State University, Texas A&M University System, Stephenville, TX, USA
[2]School of Computing and Information Sciences, Florida International University, Miami, FL, USA
[3]Department of Computer Science & Engineering, RV College of Engineering, Bengaluru, Karnataka
[4]Department of Computer Science & Engineering, Siddaganga Institute of Technology, Tumakuru, India
[5]Global Women's Health, University of North Carolina, Chapel Hill, NC, USA

Corresponding Authors: Thejas G.S. (e-mail: sadashiva@tarleton.edu), Rameshwar Garg (e-mail: ramgarg52@gmail.com).

ABSTRACT Feature selection has emerged as a craft, using which we boost the performance of our learning model. Feature or Attribute Selection is a data preprocessing technique, where only the most informative features are considered and given to the predictor. This reduces the computational overhead and improves the correctness of the classifier. Attribute Selection is commonly carried out by applying some filter or by using the performance of the learning model to gauge the quality of the attribute subset. Metric Ranked Feature Inclusion and Accuracy Ranked Feature Inclusion are the two novel hybrid feature selection methods we introduce in this paper. These algorithms follow a two-stage procedure, the first of which is feature ranking, followed by feature subset selection. They differ in the way they rank the features but follow the same subset selection technique. Multiple experiments have been conducted to assess our models. We compare our results with numerous works of the past and validate our models using 12 datasets. From the results, we infer that our algorithms perform better than many existent models.

INDEX TERMS Feature Selection, Filter, Wrapper, Hybrid, K-Means, V-Measure, Random Forest

I. INTRODUCTION

MACHINE Learning algorithms are used to extract certain information from data, with the help of statistical models. This information is used to perform various tasks without constant human intervention, by relying solely on inferences and patterns. Unfortunately, the nature and quality of the data fed to the learning algorithm determine its performance. Many times, the data might be inadequate, noisy, or erroneous, which leads to a loss in the regularity and accuracy of the predictions made by the machine. To avoid this, we have to rectify and remodel the data that the algorithm operates on. Either row correction or column correction are the possible solutions. The rows signify the input data, while the columns signify the features. A measurable property of the process under consideration is known as a feature. Each row is characterized by a vector of features and the target. The target or the class signifies the category to which that sample belongs.

Dimensionality Reduction is considered to be the best approach for column correction. Feature Selection (FS) and Feature Extraction are the two primary techniques of reducing the number of dimensions. Variable Extraction or Feature Extraction is the act of converting the given feature subset into a subset of lower dimensionality, where new features are fabricated by the combination of the available features. The number of dimensions can be minimized on applying FS as it picks a set of features from the initial set. FS can be carried out by mainly three methods: Wrapper, Embedded and Filter.

FS algorithms using the filter technique, pick features based on some score or statistical measure that is allocated to each feature. The predictor is not considered while choosing the best subset of variables in the filter approach. These algorithms are computationally less expensive and fast, but may not always give the best feature subset. FS algorithms that are classified as wrapper methods can be considered as search algorithms, where many combinations of features are

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3112169, IEEE Access

created, evaluated, and then compared with each other. The evaluation of each subset is performed with the help of the predictive model. The model runs on each subset, following which the subsets are assigned scores based on their performances. These scores are then used to pick the optimal feature subset. Many wrapper methods give better results, but cause a large overhead and may take extremely long periods of time if the feature set is extensive. Nowadays, many methods which are a combination of filters and wrappers, called Hybrids, are being devised. These Hybrid algorithms exploit the advantages of both methods while overcoming many of the disadvantages. FS algorithms employing the embedded method, choose attributes which contribute heavily to the correctness of the learning model during its creation.

The existent FS algorithms are useful but do not always prove to be extremely helpful for a wide variety of scenarios. In this paper, we propose two new FS techniques, Metric Ranked Feature Inclusion (MRFI) and Accuracy Ranked Feature Inclusion (ARFI), which can be used effectively across a variety of learning models. Our proposed algorithms are hybrids of the wrapper and filter methods and follow a two phase process. The first phase takes inspiration from the filter technique, and we assign scores for the features to rank them. For the filtering phase of MRFI, a score is assigned to each feature, after the entire dataset has been clustered based on that feature alone. We use K-Means to cluster the data and then apply a clustering metric by the name of V-Measure. ARFI involves scoring each feature based on the accuracy of a classifier, Random Forest (RF), which is evaluated with only that particular feature. Ranking the features using these techniques truly brings out their importance to the label. The next stage of the algorithm, i.e., the feature subset selection phase, aims to select those features with maximum relevance and minimum redundancy. Here, the variables are iteratively added to the optimal subset one by one, and each time, the learning model is evaluated. The recently added feature is retained or dropped depending on the calculated accuracy. The second stage behaves as the wrapper part. Both MRFI and ARFI share the same feature subset selection technique.

We validated our models with various datasets and compared our results with other standard FS techniques, including Recursive Feature Elimination (RFE). Our models outperformed RFE with every dataset and gave us positive results. The FS carried out by our models also gave superior results as compared to a variety of other FS techniques.

The paper provides a thorough review of the extensive research conducted in the past, regarding attribute selection, in Section II. A detailed explanation about our algorithms and their preliminaries is given in Section III. In Section IV, the hardware requirements and the various datasets used have been described. Section V contains discussions about our experimental outcomes. Lastly, Section VI provides an outline of the work we have carried out.

II. RELATED WORK

The efficacy of any predictor can be considerably improved by applying FS. It lessens the number of columns and thereby reduces noise. Lots of research has been conducted in this field and many survey and review papers describe various FS algorithms [1]–[3]. Several kinds of FS algorithms can be implemented, but we focus on the wrapper, filter and some hybrid methods of variable selection.

A. FILTER APPROACH

In [4], an FS technique based on correlation is proposed, in which the features are ranked based on the extent of redundancy between the attributes and their predictive capability. Kira and Rendell created the FS technique called Relief [5]. In this algorithm, weights are allocated to every variable, and KNN is used to modify the weights. Almuallim and Dietterich developed another extremely famous algorithm by the name of FOCUS [6]. This algorithm conducts a comprehensive check of all feature subsets and then finds the minimal subset that can provide accurate labeling of the training data.

Koller and Sahami [7] proposed a method which involves the elimination of a predecided number of features using backward elimination coupled with cross entropy. In [8], Liu and Setiono have implemented a method which uses random sampling to search for all feature subsets. Minimum description length of a feature subset, as the evaluation metric, was proposed by Pfahringer [9]. He makes use of simple decision tables to add or remove features. In [10], a new method of FS based on Synonym Merge, Part Of Speech and Contribution Value is used to classify Chinese text. The FS model in [11] works on the principle of multi-objective mutual information. It considers both redundancy and relevance to the class. It makes use of NSGA, which is a multi-objective search algorithm.

In [12], the author presents a unique method of reusing the discarded features after applying FS. The multitask method of learning is used to provide extra information to the classifier through the model's output. Franklin and Vasudevan [13] propose a method by the name of Highly Correlated FS (HCFS). HCFS initially sets the pertinence threshold, then finds associations among feature pairs and also among features and classes. The algorithm excludes uncorrelated features by building a tree. The feature tree is partitioned based on the relevance threshold. From this partitioning, the best feature cluster is then selected.

An FS approach, namely GClust, using interquartile range and clustering [14], has been proposed. Initially, the genes that correctly predict the classes for the inputs are chosen. The remaining genes are then clustered based on their similarity, and genes with the highest ranks are picked with the help of the Lasso method. On combining this with the initial subset, the final optimal feature subset is obtained. Kononenko [15] proposed an extension of the RELIEF model and called it ReliefF. The extension is handy as it can deal with noisy, incomplete data. Moreover, it can handle

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3112169, IEEE Access

multi-class datasets effectively. In [16], the author proposes a method based on maximum weight but minimum redundancy. The weight of a feature denotes its importance, and by using this method, we can find the subset which is most beneficial and also least correlated. Hall and Smith propose a new FS method hinged on another correlation based heuristic that can be used for the selection of a proper subset [17].

A method named INTERACT is proposed in [18] where the feature interaction is taken into consideration. Certain features may not be very relevant to the target when considered separately but might be extremely important when considered with other features. This dependency on other features is the concept of feature interactions. Irreducibility is an intrinsic character of feature interactions, which is not considered by most FS algorithms. In [19] the author proposes a model which makes use of Partial Least Squares and a decomposition technique. This model is applied to sets of two class subproblems, one versus one and one versus rest.

A comprehensive study of various statistical methods like Pearson's Coefficient and Correlation Criteria that are used to filter data, and their mathematical implementations are described in [20]–[22].

B. WRAPPER BASED APPROACH

We discuss several wrapper techniques that have been presented. In [23], the author proposes seven techniques to pick an optimal set of features. The first method uses expected probability of error. The second method chooses more features with minimum correlation using the initially picked features. The third approach is to check which feature can accurately distinguish two classes, pick the feature and repeat. The fourth is to perform Principal Component Analysis. The fifth is a small modification of the fourth, omitting those with smaller contributions. The sixth method chooses the features that make the most significant contributions to the eigenvectors. The seventh method is a mixture of the first two. A branch and bound solution is proposed for the FS problem in [24]. The algorithm begins with an unfilled subset and adds one feature at a time after gauging them. In [25], the author proposes a method to perform Sequential Floating Forward Selection in which backtracking is used to exclude variables.

The use of a Genetic Algorithm (GA) to conduct feature subset selection was initially proposed by many, but notably by [26], [27]. FS, by extending the GA, has been performed by [28]. The author proposes a wrapper method [29] to rank the features and then select them. Incremental Ranked Usefulness is used for the ranking process. Sharma *et al.* propose a new method to select features that may perform weakly when considered individually but work well with other genes [30]. The genes are divided into a small subset of size h, and then further divided into smaller informative subsets of size r. These smaller subsets are iteratively merged into a bigger, more informative subset of features.

In [31], the author proposes a method called SET-Gen, to create multiple feature subsets, with the help of a GA, along with a wrapper evaluation function. They are then evaluated using 10 fold cross validation. Caruana and Freitag [32] and John, Kohavi and Pfleger [33] evaluate several wrapper methods, which make use of hill climbing, like SLASH, Backward Stepwise Elimination, Forward Selection, Backward Elimination and Forward Stepwise Selection. In [34], a novel method of FS is proposed, which makes use of the Bat Algorithm and the Optimum-Path forest. In [35], the author proposes two novel algorithms. The first is based on the ant lion optimization operators, and the second is based on using the continuous steps of the same, as thresholds, after squashing them. In [36], an algorithm employing the Artificial Bee Colony and a perturbation parameter is presented. Mafarja *et al.* propose two methods based on Whale Optimisation Algorithm (WOA) [37]. The Tournament and Roulette Wheel selection mechanisms are used in the first approach, and Crossover and Mutation algorithms are used to improve the Whale Algorithm in the second technique.

In [38], a Binary Dragonfly optimization is proposed for FS with the help of time-varying transfer functions. In [39], the author proposes a new binary version of the Grey Wolf Optimisation technique, which is implemented for FS. Yang *et al.* propose an ensemble based wrapper method for FS [40], specifically for imbalanced class distribution. Chaouki and Saoussen [41] propose a method of FS for intrusion detection systems, using the wrapper method, enforced with the GA method. Gang and Jin Chen [42] propose a method of using wrapper methods with Support Vector Machines (SVM), namely Cosine Similarity Measure SVM to remove the unnecessary features. In [43], another SVM based technique is proposed, in which a statistics based wrapper is used in unison with the SVM for Financial Distress Identification. Lei *et al.* propose a method of FS for object based image classification [44]. Their model uses a novel wrapper technique with the help of polygon based cross validation.

C. HYBRID APPROACHES

A hybrid method of classification, which uses Modified Information FS and Modified Binary Cuckoo Search is proposed in [45]. In [46], a new method, namely class dependent density based feature elimination is proposed. It uses a measure called diff-criterion to rank the features and then perform a feature subset selection on the ranked features. In [47], the author proposes a method of FS, on the basis of ranking them initially, and then selecting a subset of attributes. The feature ranking is performed on the basis of the AUC of their decision tree model. The features are then selected based on a new logical algorithm.

In [48], the authors give us another method for FS using WOA; this time, a hybrid model. The model is based on WOA combined with simulated annealing. In [49], Hu *et al.* propose a method to select features for short term load forecasting. They implement a filter method, Partial Mutual Information followed by the firefly algorithm, which is the wrapper portion. Basant and Namita propose a method of FS [50], on the basis of Rough Set Theory and Informa-

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3112169, IEEE Access

tion Gain, which is then applied for Sentimental Analysis. A hybrid PSO is proposed in [51] by developing a new local search technique and has been named HPSO-LS. The authors of [52] propose a new technique to extract features by building a hybrid of SVM and K-Means algorithms. A new FS algorithm called TRSFFQR is developed and proposed in [53]. TRSFFQR, which stands for Tolerance Rough Set FireFly based Quick Reduct, is used for FS in MRI Brain Images classification. The techniques that have been applied are evident from the name.

Two new algorithms, PSO Relative Reduct and PSO Quick Reduct, have been proposed as FS algorithms for medical datasets in [54]. A thorough method of FS is proposed in [55], where Weighted Least Squares Twin SVM is used as a classification technique. SFS is used as the search strategy, and finally, correlation FS is used to gauge the weight of every attribute. More recent developments in the field of FS are reflected in [56], [57]. These approaches also combine unsupervised learning algorithms for the filter phase and supervised learning algorithms for the wrapper phase, to produce robust FS algorithms. In [58], Faker and Dodgu propose a hybrid method of ranking the features by applying a clustering algorithm followed by validating the clusters using homogeneity and then selecting a subset from the ranked features. A new approach, which makes use of the ReliefF algorithm followed by optimal feature subset selection with the help of SVM, is presented in [59]. Wang and Feng [60] proposed a method in which two feature subsets are created using two optimal filters. A union operation based on feature weights is developed to consolidate the two subsets. High quality clusters can be produced by hierarchical agglomerative clustering, without the total number of clusters. Hence, a predetermined threshold is used with hierarchical agglomerative clustering to produce the final feature subset.

Most of the existing FS algorithms are not extremely useful when they need to be applied to a variety of datasets and do not always give excellent results for various classifiers. Our aim is to propose FS techniques, which can be used in a wide variety of scenarios. The Hybrid method of FS has been identified as a method which incorporates the benefits of Filter and Wrapper techniques, by ranking the attributes based on their relevance and then picking a variable subset with the least amount of redundant features. Furthermore, from Section V it is clear that our proposed methodologies perform considerably better than their existent counterparts, thus exhibiting the need for better FS algorithms.

III. PROPOSED APPROACH

Here, we explain the necessary background to understand our proposed algorithms and then explain them in detail.

A. PRELIMINARIES

Here, we describe the various algorithms and metrics that we make use of in the proposed algorithms. In the feature ranking step of MRFI, we make use of K-Means and V-Measure. We make use of RF for the feature ranking stage of ARFI and the entire feature subset selection stage.

1) K-Means

K-Means falls under the category of unsupervised machine learning and is a clustering algorithm. It is used to segregate samples into the best suited group on the basis of the information already available to the algorithm. K unique clusters or groups are created such that they are sufficiently far apart from each other spatially. The distance is measured in Euclidean Distance so that clear and valid results are rendered when information is mined from them. Centroids are the centers of clusters, and data is iteratively categorized into clusters based on a data point's distance from the centroids. The most optimal solution for all the points is found iteratively as:

1) K data points are randomly chosen as centroids, with smart initialisation.
2) The distance between every point in the data set and the K randomly chosen centroids are calculated.
3) Each point is allocated to the closest cluster, on the basis of the distances calculated.
4) Centroids are reassigned by finding the average of all data points in a cluster.
5) If the centroid changes, then the process is redone from the step where the centroids are calculated until all the centroids remain the same. The clustering is complete when the centroids do not change their positions.

Mathematically, K Means seeks to reduce the squared error (objective) function. It is described below:

$$J = \sum_{a=1}^{m} \sum_{b=1}^{n} (||x_a - v_b||)^2 \tag{1}$$

Where, $||x_a - v_b||$ is the Euclidean Distance between a centroid, v_b, and a point, x_a, iterated over m points in the a^{th} cluster, for all the n clusters [61].

2) V-Measure

It is used to evaluate external clusters based on conditional entropy. It measures the goodness of the completeness and homogeneity of a cluster. Their harmonic mean is the V-Measure score of a cluster. Homogeneity of a cluster is satisfied when all the samples of a cluster are in the same, unique class. Completeness is satisfied when all the data points belonging to a single class are a part of the same cluster.

For a mathematical definition, let us consider a dataset comprising of N data points. Let these data points be partitioned into some classes, $P = \{p_x | x = 1, \ldots, m\}$ and some clusters, $Q = \{q_y | i = 1, \ldots, m\}$. The contingency table is denoted as T. This table represents the clustering solution, such that $T = \{t_{xy}\}$. Here, t_{xy} symbolizes the number of samples that are elements of the cluster q_y and members of class p_x. Let homogeneity and completeness be

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3112169, IEEE Access

represented as H and C respectively [62].

Then V-Measure is given by:

$$V_\beta = \frac{(1+\beta) \times H \times C}{(\beta \times H) + C} \tag{2}$$

Homogeneity, H, can be defined as:

$$\begin{cases} 1 & if F(P,Q) = 0 \\ 1 - \frac{F(P,Q)}{F(P)} & else \end{cases} \tag{3}$$

where,

$$F(P,Q) = -\sum_{q=1}^{|Q|}\sum_{p=1}^{|P|}\frac{t_{pq}}{N} log \frac{t_{pq}}{\sum_{p=1}^{|P|} t_{pq}} \tag{4}$$

$$F(P) = -\sum_{p=1}^{|P|}\frac{\sum_{q=1}^{|Q|} t_{pq}}{m} log \frac{\sum_{q=1}^{|Q|} t_{pq}}{m} \tag{5}$$

Completeness, C, can be defined as:

$$\begin{cases} 1 & if F(Q,P) = 0 \\ 1 - \frac{F(Q,P)}{F(Q)} & else \end{cases} \tag{6}$$

where,

$$F(Q,P) = -\sum_{p=1}^{|P|}\sum_{q=1}^{|Q|}\frac{t_{pq}}{N} log \frac{t_{pq}}{\sum_{q=1}^{|Q|} t_{pq}} \tag{7}$$

$$F(Q) = -\sum_{q=1}^{|Q|}\frac{\sum_{p=1}^{|P|} t_{pq}}{m} log \frac{\sum_{p=1}^{|P|} t_{pq}}{m} \tag{8}$$

3) Random Forest

The Random Forest can be classified under supervised learning. It is an ML algorithm which uses ensemble learning. In ensemble learning, we combine the same or different types of algorithms several times, to create a more robust prediction model. RF uses multiple decision trees and is called a forest for the same reason. It can be used for Classification and Regression.

The RF Classifier randomly picks a certain number of features from the entire database. A decision tree is then built using these features. A large number of trees are constructed in the same way, each selecting a random attribute subset of equal size. Once the forest has been created, each tree predicts the category to which the record belongs. The record is allocated to the class with most number of votes.

B. METRIC RANKED FEATURE INCLUSION (MRFI)

MRFI is a two stage process, the first of which is ranking the features, and the next stage is choosing the best attribute subset from the ranking. The ranking stage is carried out by employing K-Means and V-Measure. We split the entire dataset into training and testing datasets in the ratio of 4:1 with the standard scikit learn libraries [63]–[65]. Another dataframe to store the features in their ranked order is declared, with two columns, Name and Importance. The model selects a feature from the entire feature set, and K-Means clustering is performed, using only that feature and the target. The number of classes determine the value of K. After clustering the training data, we find the V-Measure Score of the clustering. V-Measure, being the harmonic mean of completeness and homogeneity, gives us a good understanding of the quality of the clustering. The obtained score is assigned as the importance of each feature. The entire process is carried out for each feature individually. The features, along with their importance, are then stored in the dataframe. That dataframe is then sorted to obtain a feature ranking, from most importance to least importance.

The feature subset selection stage is performed using the ranking of the features. We devise a novel algorithm for this process. The first feature in the ranking is taken, and then the accuracy of the RF classifier is calculated. The next feature is combined with the other features in the optimal feature subset from the feature ranking, and the accuracy of the same classifier is recomputed. If the accuracy increases, we retain the feature in the optimal subset. On the other hand, if the accuracy decreases, we drop the attribute from the optimal feature subset. This process is carried out iteratively for all the attributes, to obtain the final, optimal feature subset.[1]

C. ACCURACY RANKED FEATURE INCLUSION (ARFI)

Just like MRFI, ARFI also involves ranking the features and then choosing the best attribute subset from those ranked features. To rank the features, a feature is taken, and the accuracy of the random forest classifier is computed. The importance of the feature is assigned with the obtained accuracy. We carry out this process for each feature, one at a time and then add each feature with its importance to a new dataframe called Features. This dataframe has two columns, Name and Importance. The dataframe is then sorted as per the importance of the attributes, in descending order.
The next phase is the same as the one used in MRFI.

This two stage process is followed to obtain the most optimal attribute subset. The relevance of every feature is computed by ranking them and this helps in picking the most important features. Our feature subset selection method is also a novel algorithm to choose attributes from the feature ranking. This subset selection method helps us to pick features with lesser redundancy, as it evaluates the subset with the classifier to check the performance. Often, some features

[1]The GitHub link for our MRFI and ARFI packages is available at https://github.com/thejasgs/MRFI-ARFI

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3112169, IEEE Access

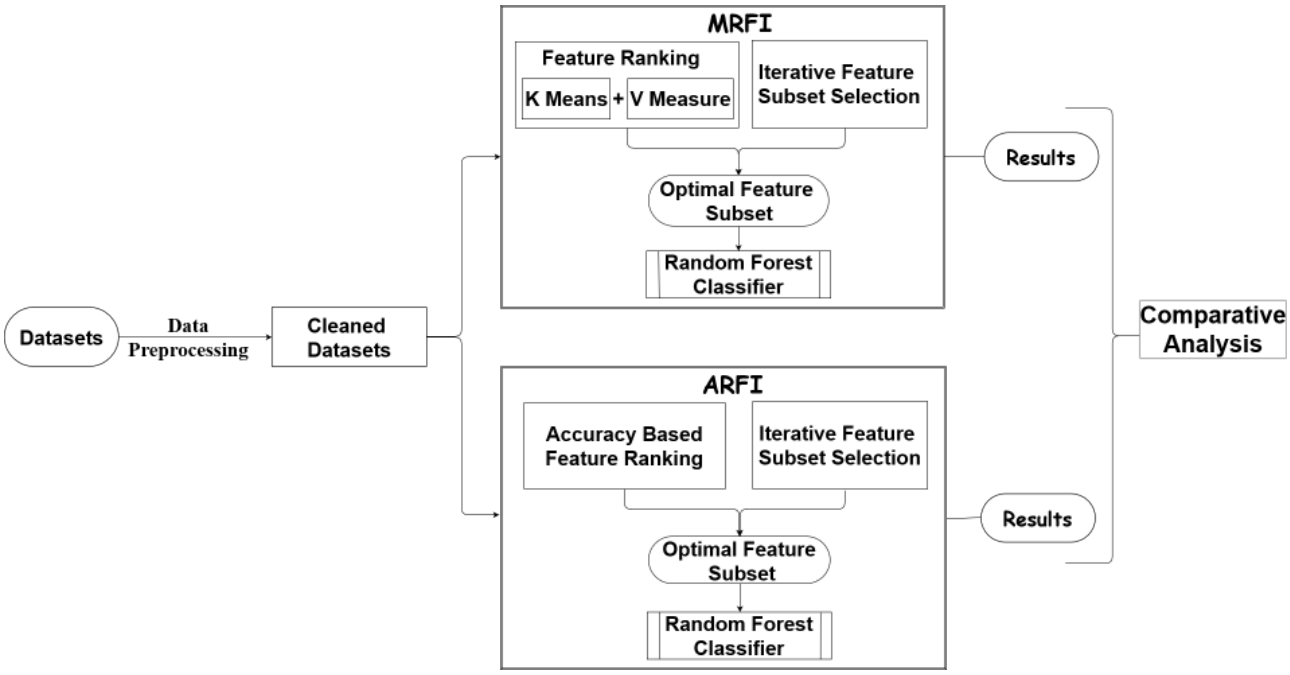

FIGURE 1: A Flowchart Depicting the Flow of Various Experiments, While Including the Framework of our Proposed Models

may not be highly relevant when considered individually but may perform really well when considered in unison with other features. Our approaches take these factors into account and give us the subset where the features perform extremely well together and are not very redundant. Throughout all the experiments spreading across the 12 datasets, the parameters of the proposed models have remained the same and have not been tuned. This has been performed in order to exhibit the models' capabilities on a level scale. We present the diagrammatic flow of our proposed approach in Fig. 1.

The filter phase of MRFI has a time complexity of $O(n * k * d * i)$ where n is the number of data samples, k is the number of clusters, d is the attribute count and i denotes the number of iterations that occur. When it comes to binary classification, the value of k is fixed as 2 and the time complexity then becomes $O(n*d*i)$. For multiclass datasets, the value of k is the number of classes, which is generally a small value. The time complexity of the learning stage of the RF classifier is $O(n * logn * d * t)$ where t is the number of trees. Since we have fixed the number of estimators for the entire process, the time complexity of the ranking phase of ARFI is $O(n * logn * d)$. The time complexity of the subset selection phase, which is common to both ARFI and MRFI, is $O(n*logn*d)$ for the best case and $O(n*logn*d^2)$ when the worst case is considered.

IV. EXPERIMENT

In this section, we give the hardware description for our experiment. We also give details about the datasets used and how we cleaned them.

A. EXPERIMENTAL SETUP

All the experiments have been implemented in Python. We made use of an Intel i7 8 Core CPU which has a 16GB RAM and the Flounder Server (AMD Opteron Processor 6380 with 64 cores and 504GB RAM).

Dataset	Feature Count	Class
UNSW - NB15	47	Binary, Multiclass (9)
Abalone	8	Multiclass (28)
Avazu	16	Binary
Breast Cancer	10	Binary
Criteo	39	Binary
Heart Disease	13	Multiclass (5)
Ionosphere	34	Binary
Iris	4	Multiclass (3)
Lung Cancer	56	Multiclass (3)
Lymphography	18	Multiclass (4)
Talking Data	9	Binary
Gene Expression Cancer RNA-Seq	20,531	Multiclass (5)

TABLE 1: An Outline of the Various Datasets Used in Our Experiments

B. DATASETS

For our experiments, we make use of 12 datasets in total. Firstly, we use the UNSW-NB15 Dataset [66], a standard dataset for Intrusion Detection Systems. Three click fraud datasets, Avazu [67], Criteo [68] and Talking Data [69] have also been experimented on. The remaining datasets are standard, benchmark datasets which are available in the UCI Machine Learning Repository [70]. This repository is a store-house of databases, created by David Aha and other graduate students from UCIrvine. We make use of the Abalone, Breast Cancer [71], Heart Disease, Ionosphere, Gene Expression

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3112169, IEEE Access

Cancer RNA-Seq [72], Iris, Lung Cancer and Lymphography datasets to evaluate and validate our models. The UNSW dataset can be used as a binary dataset and a multiclass dataset. The Abalone, Heart Disease, Iris, Gene Expression Cancer RNA-Seq, Lung Cancer and Lymphography datasets fall under the category of multiclass datasets, whereas the Breast Cancer, Ionosphere, Avazu, Talking Data and Criteo datasets fall under the category of binary datasets. Details about the number of features, and the various types of datasets considered, are shown in Table 1.

C. DATA PREPROCESSING

UNSW - NB15

The UNSW dataset initially has 2540047 rows. We use 43 out of the 47 features for classification. There are two label columns, 'attack_cat' for multiple classification and 'Label' for binary classification. There are 9 types of attacks which are considered for multiclass classification. We use the entire UNSW dataset, for which we append the four datasets given in [85]. Then, we manually assign the column names mentioned in the information file of UNSW dataset. We drop the first four columns, 'scrip', 'sport', 'dstip' and 'dsport', as they are just identification numbers and are not significant. The column 'attack_cat' is used to perform multiclass classification. All the NaN values in the 'attack_cat' column are dropped to retain only the attack types. The remaining NaN values in the dataset are filled with zeroes as they are all numerical values which represent some count. The same classes occur multiple times in the 'attack_cat' column, with different names and white spaces. These white spaces are stripped, and the names are standardized. The 'ct_ftp_cmd' column has string representations of numbers. We convert them back to numbers and encode the 'service', 'proto' and 'service' columns. We normalize the dataset using the Standard Scaler. For the binary classification using the same dataset, we do not consider the 'attack_cat' column and all the remaining NaN values are dropped. The remaining preprocessing of this dataset is carried out in the same way as that of the multiclass classification version.

Abalone

The Abalone dataset has a total of 4177 entries, categorized into 28 classes. All 8 features are used. We perform encoding to convert the column with genders into numbers.

Avazu

The Avazu dataset is a click fraud dataset recorded over 10 days. We split the column called 'hour of click' into three separate columns. The class ratio of the 200 million rows of the original dataset is maintained when the rows are reduced to 1 million rows. The original dataset consisted of 16 features and one label column.

Breast Cancer Wisconsin (Original)

This consisted of 699 rows before the empty values were dropped. There are 10 features that we make use of, along with one target column. We replace the string representation of numbers with the actual numbers in some columns.

Criteo

The Criteo dataset is another click fraud dataset which we use to validate our models. It has 756554 rows and 39 features. We dropped all the rows which had NaN values.

Cleveland Heart Disease

The dataset providing information about Heart Diseases in Cleveland has 303 rows. It consists of 13 features and one target column with 5 classes. We dropped all the rows which contained undetermined values and replaced the string representations of all the numbers with the actual numbers.

Ionosphere

The Ionosphere dataset has 351 rows and 34 features that we use. We perform label encoding on the target column and then conduct our experiment.

Iris

The famous Iris dataset consists of only 4 features and only 150 rows. The rows are classified into three labels. We perform label encoding on this dataset. We also shuffle the entire dataset to get a good mix of all the classes.

Lung Cancer

Another famous dataset called the Lung Cancer dataset has been used. It is composed of only 32 rows of data but has 56 features. The number of rows is further reduced after dropping the missing values. It has 3 classes into which the rows can be categorized. We run the classification algorithms after the data is shuffled to get a good mix of all three classes.

Lymphography

The Lymphography dataset is composed of 148 rows and has 18 features. The label column has 4 classes. We perform random shuffling to get a uniform distribution of classes to help the machine learn effectively.

Talking Data

Talking Data has a million rows, and they are denoted with 9 features. The column called 'attributed time' is dropped because it consisted of a large number of NaN values. The attribute 'click time' composed of the time-stamps, is split into four new attributes: 'day', 'hour', 'minute', and 'second'. We randomly select 1 million observations from around 200 million, but the class ratio is kept constant [73], [74].

Gene Expression Cancer RNA-Seq

The Gene Expression Cancer RNA-Seq consists of 801 rows, and a massive aggregate of 20,531 variables. The elements are categorised into five classes. We replaced the string class names with numerical values, by performing label encoding.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3112169, IEEE Access

V. RESULTS AND DISCUSSIONS

We present our results and make comparisons with previous work after giving details about the classifier, the various metrics used, and our method of analysis.

A. BASE CLASSIFIER

We use RF as our base classifier to carry out multiclass and binary classification. A thorough working of the RF classifier has been described in Section III.

B. EVALUATION METRICS

To gauge the efficacy of our classifier, we employ specific metrics, namely, Recall (Rec), Accuracy (Acc), Precision (Prec) and F1 Score. Furthermore, we evaluate the AUC score for binary classification. These evaluation metrics make use of: T_{Posi}, which represents the correctly predicted positives values, T_{Negi}, which describes the correctly obtained negative values, F_{Posi}, representing the wrongly obtained positive values and F_{Negi} describing the wrongly predicted negative values [75]. The metrics, as mentioned above, are computed with the formulae given below:

$$Acc = \frac{T_{Posi} + T_{Negi}}{T_{Posi} + F_{Posi} + F_{Negi} + T_{Negi}} \tag{9}$$

$$Prec = \frac{T_{Posi}}{T_{Posi} + F_{Posi}} \tag{10}$$

$$Rec = \frac{T_{Posi}}{T_{Posi} + F_{Negi}} \tag{11}$$

$$F1 \quad Score = 2 \times \frac{Rec \times Prec}{Rec + Prec} \tag{12}$$

The Receiving Operator Characteristic (ROC) curve is a plot of the T_{Posi} rate against the F_{Posi} rate. These values are plotted for all possible cut-off values. A popular metric used to cross check the above metrics and avoid overfitting and underfitting, is the Area Under the Receiving Operator Characteristic Curve (AUC). It can also be interpreted as the average T_{Posi} rate for all F_{Posi} rates [76].

C. METHOD OF ANALYSIS

To compare the above metrics and validate our models, we follow a standard order. After cleaning the dataset, we run the base classifier, i.e., the RF classifier without any FS and record the results. Then, we run MRFI to obtain an optimal feature subset. We rerun the RF classifier with this new feature subset and record the results. We conduct the same experiment with ARFI. Once our algorithms have been evaluated, we perform RFE on the original dataset and compute the above metrics. Various works [77]–[80] regarding FS, have considered RFE as a benchmark FS algorithm, thereby making it a yardstick for comparison. Moreover, RFE behaves like a Hybrid FS model as it ranks the attributes based on feature importances, and then recursively eliminates the worst feature according to the ranking. The feature elimination takes place after it evaluates the entire subset with the classifier. It has also proven to be extremely efficient in the past. RFE requires an external parameter which tells it how many features are to be considered. The parameter is given based on the number of features considered by our models. Next, we examine the results of the original dataset, our algorithms, RFE and other contemporaries. Our models have performed exceedingly well, as can be seen from the results. Furthermore, we have also conducted statistical tests to support and verify the results obtained from our experiments. We have presented the results of the experiments and the statistical analysis in the form of tables and plots below.

D. DISCUSSIONS

Unlike RFE, our models do not need to know the number of features beforehand. Our FS algorithms iteratively add features and do not need a fixed, minimum or maximum number of features. Figure 3 depicts the same. Only feature subsets with a minimum accuracy of 83 percent have been shown in the figure. Each point denotes a feature subset that is being evaluated. As can be seen, subsets are evaluated immediately after their creation. Only if the performance of the learning model does not decrease, the most recently included feature is considered in the final subset. The occurrence of multiple points of the same FS algorithm, along one vertical line, represents subsets that do not perform as well as their immediate predecessors. This happens due to redundancy. Even though the attributes are ranked by their relevance, the redundancy between them may cause the subset to underperform. ARFI and MRFI overcome this issue in the second stage of their algorithms. For the Avazu dataset, ARFI considers 8 features, whereas MRFI considers 9, as the addition of any more features reduces the accuracy of the learning model.

From Table 2, it is clear that both MRFI and ARFI give us outstanding results for all six multiclass datasets. Both the proposed models outperform RFE and even tend to improve the performance of the predictor.

When compared to each other, ARFI performs better than MRFI in three of the datasets, namely UNSW, Heart Disease and Lymphography. In the other three datasets, they both give similar levels of performance. From our results, it appears that ARFI considers lesser redundant features, as it always selects lesser or equal number of features compared to MRFI, and performs better with those features.

For the binary datasets (Table 3), ARFI gives good results when used with the Breast Cancer, Ionosphere and Avazu datasets. The accuracy and the AUC of the classifier after applying ARFI fall for the UNSW, Talking Data and Criteo datasets. The reason for this can be explained after understanding the results proposed in [81]. When there is no additional informational being extracted with the help of FS, the evaluation metrics might not increase and may even get negatively affected. Furthermore, when the sample size is big enough, the classifier can get trained well enough to predict

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3112169, IEEE Access

Dataset	FS Algorithm	No. Of Features	Accuracy	Precision	Recall	F1 Score
UNSW - NB15	None	43	89.260	89.620	89.260	88.890
	RFE	10	89.473	89.670	89.470	88.010
	ARFI	10	90.108	90.890	90.110	88.860
	MRFI	14	90.068	91.030	90.070	88.820
Abalone	None	8	24.521	23.050	24.520	22.890
	RFE	1	17.584	17.830	17.580	17.420
	ARFI	1	25.120	22.520	25.120	21.970
	MRFI	1	25.120	22.520	25.120	21.970
Heart Disease	None	13	46.667	34.350	46.670	39.310
	RFE	6	40.000	29.540	40.000	33.510
	ARFI	5	51.667	48.720	51.670	48.760
	MRFI	6	48.333	41.370	48.330	43.890
Iris	None	4	93.333	93.330	93.330	93.330
	RFE	2	93.333	93.330	93.330	93.330
	ARFI	2	96.667	96.920	96.670	96.580
	MRFI	2	96.667	96.920	96.670	96.580
Lung Cancer	None	56	50.000	37.500	50.000	40.000
	RFE	29	50.000	38.890	50.000	43.330
	ARFI	10	83.333	88.890	83.330	82.220
	MRFI	29	83.333	88.890	83.330	82.220
Lymphography	None	18	80.000	80.190	80.000	79.720
	RFE	11	86.667	77.330	80.000	78.460
	ARFI	6	93.333	94.290	93.330	93.390
	MRFI	11	90.000	90.050	90.000	89.920
Gene Expression Cancer RNA-Seq	None	20,531	98.757	99.260	98.730	98.980
	RFE	95	94.409	95.890	94.110	94.810
	ARFI	8	99.379	99.620	99.380	99.490
	MRFI	95	99.378	99.620	99.350	99.480

TABLE 2: Experimental Results of Classification on Multiclass Datasets

values more accurately on its own. The effect of FS also depends on the features and the degree of correlation between them. The classifier used can also affect the improvement in performance after applying FS, as some datasets tend to perform better with particular classifiers. MRFI gives us similar results, as it performs well on the same datasets as ARFI.

Research Work	Classifier	Accuracy(%)
Tama & Primartha [82]	RF	95.50
	Multilayer Perceptron	83.50
Moustafa & Slay [83]	Naive Bayes	79.50
	Expectation Maximisation	77.20
	Linear Regression	83.00
Belouch *et al.* [84]	Naive Bayes	80.04
	RepTree	87.80
	Decision Tree	86.13
	Random Tree	86.59
	Artificial Neural Network	86.31
Salaf *et al.* [85]	Naive Bayes	74.19
	RF	97.49
	Decision Tree	95.82
	SVM	92.28
Al et. al [86]	Deep Learning	98.99
Faker & Dogdu [58]	Gradient Boosted Tree	97.92
	RF	98.86
	Deep Neural Network	99.19
Our Work	**RF**	**99.94**
	RF + MRFI	99.92
	RF + ARFI	99.92

TABLE 4: Comparison of Binary Classification with Previous Works using UNSW-NB15 Dataset

ARFI tends to give us superior results when compared with MRFI for four datasets, considering the accuracy. In the Breast Cancer and Ionosphere datasets, they both render similar levels of accuracy. MRFI gives a better AUC value for the Avazu dataset but falls behind ARFI in all the other datasets.

Research Work	Classifier	Accuracy(%)
Belouch *et al.* [84]	Naive Bayes	73.86
	RepTree	79.20
	Artificial Neural Network	78.14
	Random Tree	76.21
Our Work	RF	89.26
	RF + MRFI	90.07
	RF + ARFI	**90.10**

TABLE 5: Comparison of Multiclass Classification with Previous Works using UNSW-NB15 Dataset

Figure 2(a) portrays the change in accuracy after performing FS on the Heart Disease, Lung Cancer and Lymphography datasets. The larger variations in accuracy are seen in this figure. On the other hand, Fig. 2(b) represents the smaller changes in accuracy observed on applying FS on the Iris, UNSW and Abalone datasets. From the plots, we infer that ARFI and MRFI perform considerably better than RFE, as the changes in accuracy are preferable.

Figures 2(c) and 2(d) depict changes in accuracy on performing FS on various datasets. It is evident that our models perform satisfactorily when compared to RFE for most datasets.

Both our models' results considerably outdo the results obtained after applying RFE. Now, we compare our models and their performance with other models previously applied

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3112169, IEEE Access

Dataset	FS Algorithm	No. Of Features	Accuracy	Precision	Recall	F1 Score	AUC
UNSW - NB15	None	43	99.939	99.940	99.940	99.940	99.513
	RFE	19	99.894	99.894	99.894	99.894	98.923
	ARFI	19	99.924	99.924	99.924	99.924	99.261
	MRFI	29	99.918	99.20	99.20	99.20	99.920
Breast Cancer	None	10	99.270	99.280	99.270	99.270	99.057
	RFE	9	98.540	98.570	98.540	98.530	98.113
	ARFI	9	99.999	99.999	99.999	99.999	99.999
	MRFI	7	99.999	99.999	99.999	99.999	99.999
Ionosphere	None	34	94.366	94.370	94.370	94.370	93.238
	RFE	21	92.958	92.900	92.960	92.910	90.857
	ARFI	21	95.775	95.760	95.770	95.740	94.238
	MRFI	16	95.774	94.238	95.760	95.770	95.740
Talking Data	None	9	95.173	95.170	95.170	95.080	91.673
	RFE	6	94.788	94.780	94.790	94.680	91.068
	ARFI	3	95.121	95.150	95.120	95.010	91.360
	MRFI	6	94.223	94.180	94.220	94.110	90.401
Criteo	None	39	73.567	72.330	73.570	70.320	62.420
	RFE	3	67.043	63.480	67.040	63.850	55.903
	ARFI	3	70.270	67.670	70.270	67.210	59.381
	MRFI	3	70.140	67.290	71.140	65.750	57.628
Avazu	None	16	82.896	78.210	82.900	78.190	54.388
	RFE	8	82.993	78.080	82.990	77.510	53.137
	ARFI	8	83.328	79.260	83.330	78.130	54.040
	MRFI	10	83.295	79.170	83.290	78.280	54.322

TABLE 3: Experimental Results of Classification on Binary Datasets

on the UNSW-NB15 dataset. We also compare them with the Talking Data and Ionosphere Datasets.

Table 4 compares various results for the UNSW dataset for Binary Classification. It compares the results on the basis of accuracy. Our model obtains the highest accuracy and gives the best performance. Another noticeable fact, is that our RF performs much better than other RF models that have been used before. This is traced to the method of preprocessing the UNSW dataset, including normalization.

Comparisons of our work with previous work concerning the UNSW dataset for multiclass classification have been shown in Table 5. Our model clearly outperforms the model proposed in [84].

Research Work	Classifier	Accuracy(%)
Liu & Zhang [87]	KNN + LS	88.32
	KNN + FS	89.18
	KNN + CS	89.59
	KNN + Lasso	87.46
	KNN + CGS	91.32
Ghaemi *et al.* [88]	J48	93.16
	3NN	92.30
	5NN	89.43
	RBF-SVM	94.58
	1NN	89.52
	J48	95.12
Our Work	RF	94.36
	RF + MRFI	**95.77**
	RF + ARFI	**95.77**

TABLE 6: Comparison of Binary Classification on Ionosphere Dataset with Previous Works

Table 6 portrays a comparison between the work performed on the Ionosphere dataset. It is evident that both our models outperform most of the other FS models. RBF-SVM and J48 give better results than our base classifier and MRFI, but ARFI outperforms both of them too.

When compared to the previous work of Qiu *et al.*, our models have higher precision, recall and F1 scores. The dataset under consideration is the Talking Dataset. These results can be seen in Table 7.

Research Work	Classifier	Prec	Rec	F1 Score
Qiu *et al.* [89]	SVM	0.896	0.877	0.876
	Naive Bayes	0.908	0.897	0.896
	ETCF	0.910	0.904	0.904
	GBDT	0.906	0.891	0.890
	RF	0.877	0.846	0.842
Our Work	RF	0.951	0.951	0.950
	RF + MRFI	0.941	0.942	0.941
	RF + ARFI	**0.951**	**0.951**	**0.950**

TABLE 7: Comparison of Binary Classification on Talking Data Dataset with Previous Works

Classifier	FS Algorithm	Accuracy (%)
Random Forest	None	46.67
	MRFI	48.33
	ARFI	51.67
Decision Tree	None	38.33
	MRFI	41.66
	ARFI	55.00
Support Vector Machine	None	48.33
	MRFI	50.00
	ARFI	48.33
K Nearest Neighbour	None	40.00
	MRFI	41.67
	ARFI	43.33
Naive Bayes	None	41.67
	MRFI	53.33
	ARFI	53.33
AdaBoost	None	45.00
	MRFI	45.00
	ARFI	51.67

TABLE 8: Performance of the Proposed Algorithms for Varying Base Classifiers on the Heart Disease Dataset

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3112169, IEEE Access

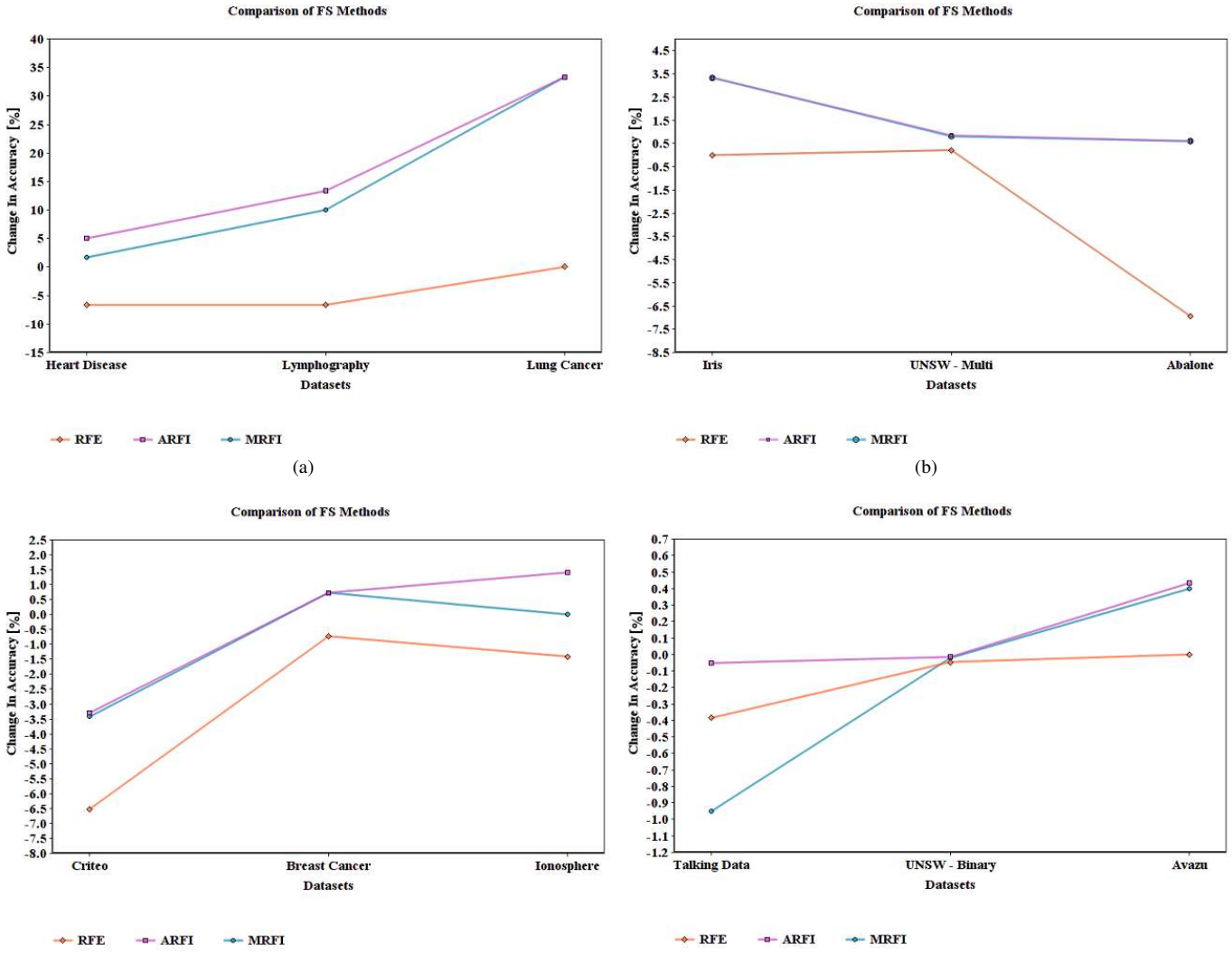

FIGURE 2: Changes in Accuracy for Different Datasets Using Three Feature Selection Models.
(a) depicts larger changes in accuracy after applying FS on the Multiclass datasets. (b) depicts smaller changes in accuracy after performing FS on the Multiclass datasets. (c) depicts greater changes in accuracy after applying FS on the Binary datasets. (d) depicts minute changes in accuracy after performing FS on the Binary datasets.

Hybrid FS techniques benefit from the advantages of the filter and wrapper approaches. Ranking the features effectively and then selecting an optimal subset is of prime importance. The ranking phase of MRFI is very effective as the score assigned to each attribute is based on the V-Measure of the cluster, a metric created to gauge the true goodness and validity of the clustering. The filter component of ARFI is also extremely unique as it incorporates the classifier in order to rank the features. This method of ranking the features based on the accuracy of classification proves to be very helpful with certain cases where FS is not generally effective. The above can be observed in Table 8, where the results of classification before and after applying MRFI & ARFI, for a variety of base classifiers, have been presented. The experiment has been carried out for the Heart Disease Dataset, and it is evident that our algorithms increase the correctness of the predictions for all types of classifiers.

Although clustering and classification are computationally heavy, ranking the attributes with their help differentiates MRFI and ARFI from their contemporaries, as the true relevance of the features is brought to light. ARFI and MRFI are even more distinguished because of their unique subset selection process. The second stage of our proposed methods is crucial as redundant variables are not included in the optimal feature subset. Some features tend to become more relevant when grouped with other attributes. Our methods account for this fact as well. In a nutshell, MRFI and ARFI perform better than the current FS algorithms because of their

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3112169, IEEE Access

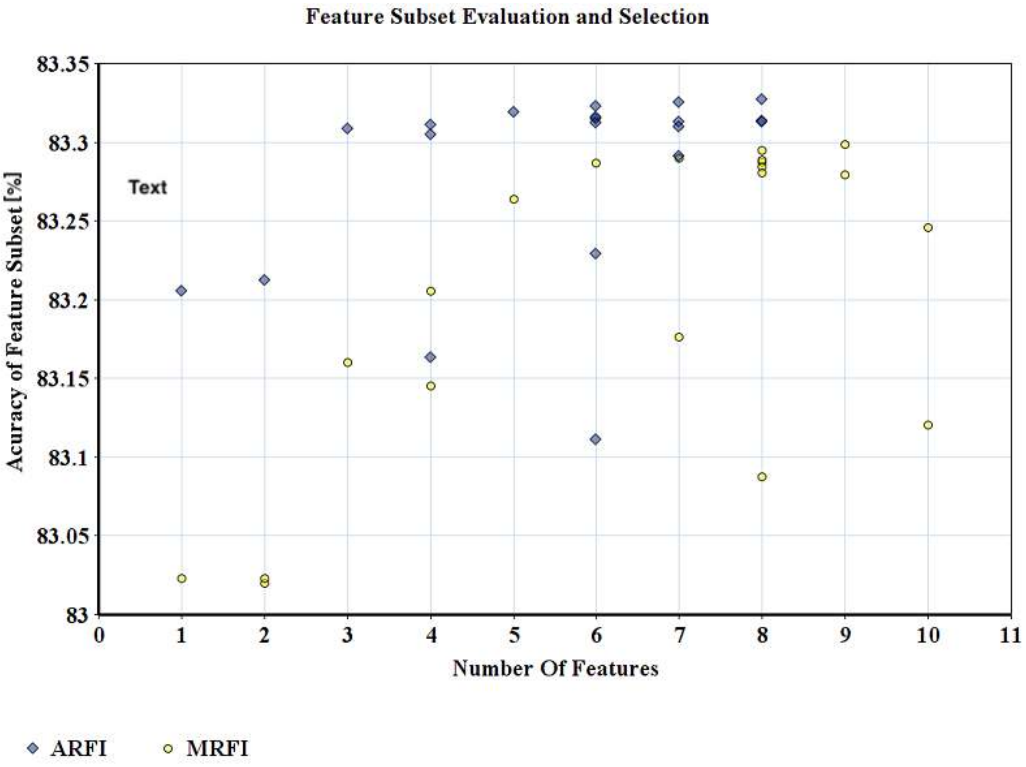

FIGURE 3: Scatter plot of the accuracy of multiple feature subsets, that were created by ARFI and MRFI, depicting the feature subset selection procedure for the Avazu dataset.

exceptional feature ranking abilities as well as their critical subset selection phase.

E. STATISTICAL ANALYSIS

To further support the comparative analysis shown in Tables 2 and 3, statistical tests were performed to gauge the performance of MRFI and ARFI in comparison with RFE. These tests were employed to confirm the significance of the results obtained. They were performed using R Studio (Version 1.3.1073).

Non-parametric tests were used to verify the significance of the predicted values obtained from the classifiers for all the datasets [90], [91]. As suggested in various articles [90]–[92], we have used non-parametric methods of statistical analysis due to the following reasons:

- They overcome the major limitation of parametric methods, where the data needs to be distributed normally.
- Non-parametric tests share the reliability of parametric tests.
- Parametric tests can reliably reject a null hypothesis only if some assumptions are not violated.

We present the results of the statistical tests for the binary and multiclass datasets below.

1) Binary Datasets

In order to compare the FS methodologies, we processed the predicted values obtained from the classifier after applying FS. To evaluate the proposed models, we compared them with existing FS algorithms. For this purpose, the requirement is to have a statistical test which is based on the chi statistic distribution for goodness of fit in data with small sample sizes. To assess the significance of the improvement after applying the proposed FS methods, McNemar's test has been employed [92], [93]. Having a non-parametric nature, the McNemar's test is applied on a 2X2 classification table, as can be seen in Table 9, to quantify the dissimilarity between the paired proportions [94].

	FS B Failed	**FS B Succeeded**
FS A Failed	N_{ff}	N_{fs}
FS A Succeeded	N_{sf}	N_{ss}

TABLE 9: McNemar's Table Depicting the Results of Two FS Algorithms

N_{ff} represents the number of times when both the FS algorithms failed to classify instances correctly, while N_{ss} denotes the opposite. Sufficient information cannot be gathered regarding the algorithms' significance from these two values, as there is no indication about the difference in their performance. The two other values, N_{sf} and N_{fs}, denote situations where one of the FS algorithms succeeded and the other failed, implying performance discrepancies.

The null hypothesis H_0 in our experiment states that there is no significant difference between the results obtained after classification using various FS algorithms. We reject the null

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3112169, IEEE Access

hypothesis if the p-value of the test is less than 0.05 and accept it if the p-value is greater than 0.05 with an alpha value of 95% and 1 degree of freedom. The results are recorded in Table 10.

Dataset	FS Algorithm vs RFE	P-Value
UNSW - NB15	MRFI	0.02737
	ARFI	0.0003692
Breast Cancer	MRFI	0.00000000000000022
	ARFI	0.00000000000000022
Ionosphere	MRFI	0.0000000000000007015
	ARFI	0.00000000003605
Talking Data	MRFI	0.00000000000000022
	ARFI	0.00000000000000022
Criteo	MRFI	0.00000000000000022
	ARFI	0.00000000000000022
Avazu	MRFI	0.00000000000000022
	ARFI	0.00000000000000022

TABLE 10: McNemar's Test: P-Values from the Comparison of FS Algorithms for Binary Datasets

Since the p-values for all the paired tests are less than 0.05, the null hypothesis H_0 is rejected for all the tests. Therefore, based on the results of the test, the performance difference between RFE and the proposed algorithms has been proved to be statistically significant.

2) Multiclass Datasets

Dataset	FS Algorithm vs RFE	P-Value
UNSW - NB15	MRFI	0.0000000000024
	ARFI	0.000000000001188
Abalone	MRFI	< 0.0001
	ARFI	< 0.0001
Heart Disease	MRFI	0.01115
	ARFI	0.004161
Iris	MRFI	0.0000005043
	ARFI	0.0000008954
Lung Cancer	MRFI	0.013112
	ARFI	0.01261
Lymphography	MRFI	0.00001576
	ARFI	0.00007844
Gene Expression	MRFI	0.00000022
	ARFI	0.00000022

TABLE 11: Friedman's Test: P-Values from the Comparison of FS Algorithms for Multiclass Datasets

To statistically verify the results obtained for the multiclass datasets, we have made use of the Friedman test as recommended by various scholarly articles [95], [96]. This test is a non-parametric approach which can be applied to multiclass datasets. The results of the test have been presented in Table 11. The null hypothesis used for this test is that the distribution of predictions made by the proposed algorithms are the same as that of RFE. The tests have been performed using the built-in functions of R [97].

As can be seen from Table 11, the p-values for all the tests are lower than 0.05 and the null hypothesis H_0 can be rejected with a confidence of 95%. It can be concluded that the difference between the results of MRFI and ARFI when compared to RFE are large enough to be statistically significant. Hence, both the proposed models outperform RFE significantly.

VI. CONCLUSION

Feature Selection is an essential tool that is used to select a feature subset using which the performance of the classifier can be improved. FS is vital as it reduces the training time of the model under consideration, reduces overfitting, and more importantly, avoids the curse of dimensionality. In our paper, we present two new FS methods, MRFI and ARFI. Both the models are hybrids of filter and wrapper methods of FS. MRFI employs K-Means and V-Measure scores to rank the features, whereas ARFI ranks the features based on the accuracy of the predictor (Random Forest). Both our methods follow the same feature subset selection technique. We compare our models with Recursive Feature Elimination, a state-of-the-art FS technique, using 12 datasets. Furthermore, we gauge their performance with the help of previous work carried out on the same datasets. We observe that our models have performed well and have given superior results. A noticeable limitation of both our algorithms, is that they have a time complexity which sits on the higher end, but this can be overcome if the computational resources are powerful. After applying FS using our proposed methods, the evaluation metrics improve drastically, and the accuracy of the random forest classifier also increases considerably, thereby overcoming the drawbacks of the current FS algorithms.

REFERENCES

[1] D. Mladenić and M. Grobelnik, "Feature selection on hierarchy of web documents," Decision Support Systems, vol. 35, no. 1, pp. 45–87, 2003.
[2] V. Bolón-Canedo, N. Sánchez-Marono, A. Alonso-Betanzos, J. M. Benítez, and F. Herrera, "A review of microarray datasets and applied feature selection methods," Information Sciences, vol. 282, pp. 111–135, 2014.
[3] G. Chandrashekar and F. Sahin, "A survey on feature selection methods," Computers & Electrical Engineering, vol. 40, no. 1, pp. 16–28, 2014.
[4] K. Pushpalatha and A. G. Karegowda, "Cfs based feature subset selection for enhancing classification of similar looking food grains-a filter approach," in 2017 2nd International Conference On Emerging Computation and Information Technologies (ICECIT). IEEE, 2017, pp. 1–6.
[5] K. Kira, L. A. Rendell et al., "The feature selection problem: Traditional methods and a new algorithm," in Aaai, vol. 2, 1992, pp. 129–134.
[6] H. Almuallim and T. G. Dietterich, "Efficient algorithms for identifying relevant features," in In Proceedings of the Ninth Canadian Conference on Artificial Intelligence. Citeseer, 1992.
[7] D. Koller and M. Sahami, "Toward optimal feature selection," Stanford InfoLab, Tech. Rep., 1996.
[8] H. Liu, R. Setiono et al., "A probabilistic approach to feature selection-a filter solution," in ICML, vol. 96. Citeseer, 1996, pp. 319–327.
[9] B. Pfahringer, Compression based feature subset selection. Österr. Forschungsinst. für Artificial Intelligence, 1995.
[10] S. Qin, J. Song, P. Zhang, and Y. Tan, "Feature selection for text classification based on part of speech filter and synonym merge," in 2015 12th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD). IEEE, 2015, pp. 681–685.
[11] M. Labani, P. Moradi, M. Jalili, and X. Yu, "An evolutionary based multi-objective filter approach for feature selection," in 2017 World Congress on Computing and Communication Technologies (WCCCT). IEEE, 2017, pp. 151–154.
[12] R. Caruana and V. R. d. Sa, "Benefitting from the variables that variable selection discards," Journal of machine learning research, vol. 3, no. Mar, pp. 1245–1264, 2003.

[13] D. F. Vinod and V. Vasudevan, "A filter based feature set selection approach for big data classification of patient records," in 2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT). IEEE, 2016, pp. 3684–3687.

[14] Z. Khan, M. Naeem, U. Khalil, D. M. Khan, S. Aldahmani, and M. Hamraz, "Feature selection for binary classification within functional genomics experiments via interquartile range and clustering," IEEE Access, vol. 7, pp. 78 159–78 169, 2019.

[15] I. Kononenko, "Estimating attributes: analysis and extensions of relief," in European conference on machine learning. Springer, 1994, pp. 171–182.

[16] J. Wang, L. Wu, J. Kong, Y. Li, and B. Zhang, "Maximum weight and minimum redundancy: a novel framework for feature subset selection," Pattern Recognition, vol. 46, no. 6, pp. 1616–1627, 2013.

[17] M. A. Hall and L. A. Smith, "Practical feature subset selection for machine learning," 1998.

[18] Z. Zhao and H. Liu, "Searching for interacting features in subset selection," Intelligent Data Analysis, vol. 13, no. 2, pp. 207–228, 2009.

[19] S. Student and K. Fujarewicz, "Stable feature selection and classification algorithms for multiclass microarray data," Biology direct, vol. 7, no. 1, p. 33, 2012.

[20] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," Journal of machine learning research, vol. 3, no. Mar, pp. 1157–1182, 2003.

[21] J. Tang, S. Alelyani, and H. Liu, "Feature selection for classification: A review," Data classification: Algorithms and applications, p. 37, 2014.

[22] J. Novaković, "Toward optimal feature selection using ranking methods and classification algorithms," Yugoslav Journal of Operations Research, vol. 21, no. 1, 2016.

[23] A. N. Mucciardi and E. E. Gose, "A comparison of seven techniques for choosing subsets of pattern recognition properties," IEEE Transactions on Computers, vol. 100, no. 9, pp. 1023–1031, 1971.

[24] P. M. Narendra and K. Fukunaga, "A branch and bound algorithm for feature subset selection," IEEE Transactions on computers, no. 9, pp. 917–922, 1977.

[25] P. Pudil, J. Novovičová, and J. Kittler, "Floating search methods in feature selection," Pattern recognition letters, vol. 15, no. 11, pp. 1119–1125, 1994.

[26] J. Yang and V. Honavar, "Feature subset selection using a genetic algorithm," in Feature extraction, construction and selection. Springer, 1998, pp. 117–136.

[27] W. F. Punch III, E. D. Goodman, M. Pei, L. Chia-Shun, P. D. Hovland, and R. J. Enbody, "Further research on feature selection and classification using genetic algorithms." in ICGA, 1993, pp. 557–564.

[28] L. J. Eshelman, "The chc adaptive search algorithm: How to have safe search when engaging in nontraditional genetic recombination," in Foundations of genetic algorithms. Elsevier, 1991, vol. 1, pp. 265–283.

[29] R. Ruiz, J. C. Riquelme, and J. S. Aguilar-Ruiz, "Incremental wrapper-based gene selection from microarray data for cancer classification," Pattern Recognition, vol. 39, no. 12, pp. 2383–2392, 2006.

[30] A. Sharma, S. Imoto, and S. Miyano, "A top-r feature selection algorithm for microarray gene expression data," IEEE/ACM Transactions on Computational Biology and Bioinformatics, vol. 9, no. 3, pp. 754–764, 2011.

[31] K. J. Cherkauer and J. W. Shavlik, "Growing simpler decision trees to facilitate knowledge discovery." in KDD, vol. 96, 1996, pp. 315–318.

[32] R. Caruana and D. Freitag, "Greedy attribute selection," in Machine Learning Proceedings 1994. Elsevier, 1994, pp. 28–36.

[33] G. H. John, R. Kohavi, and K. Pfleger, "Irrelevant features and the subset selection problem," in Machine Learning Proceedings 1994. Elsevier, 1994, pp. 121–129.

[34] D. Rodrigues, L. A. Pereira, R. Y. Nakamura, K. A. Costa, X.-S. Yang, A. N. Souza, and J. P. Papa, "A wrapper approach for feature selection based on bat algorithm and optimum-path forest," Expert Systems with Applications, vol. 41, no. 5, pp. 2250–2258, 2014.

[35] E. Emary, H. M. Zawbaa, and A. E. Hassanien, "Binary ant lion approaches for feature selection," Neurocomputing, vol. 213, pp. 54–65, 2016.

[36] M. Schiezaro and H. Pedrini, "Data feature selection based on artificial bee colony algorithm," EURASIP Journal on Image and Video Processing, vol. 2013, no. 1, p. 47, 2013.

[37] M. Mafarja and S. Mirjalili, "Whale optimization approaches for wrapper feature selection," Applied Soft Computing, vol. 62, pp. 441–453, 2018.

[38] M. Mafarja, I. Aljarah, A. A. Heidari, H. Faris, P. Fournier-Viger, X. Li, and S. Mirjalili, "Binary dragonfly optimization for feature selection using time-varying transfer functions," Knowledge-Based Systems, vol. 161, pp. 185–204, 2018.

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

[40] P. Yang, W. Liu, B. B. Zhou, S. Chawla, and A. Y. Zomaya, "Ensemble-based wrapper methods for feature selection and class imbalance learning," in Pacific-Asia Conference on Knowledge Discovery and Data Mining. Springer, 2013, pp. 544–555.

[41] C. Khammassi and S. Krichen, "A ga-lr wrapper approach for feature selection in network intrusion detection," computers & security, vol. 70, pp. 255–277, 2017.

[42] G. Chen and J. Chen, "A novel wrapper method for feature selection and its applications," Neurocomputing, vol. 159, pp. 219–226, 2015.

[43] H. Li, C.-J. Li, X.-J. Wu, and J. Sun, "Statistics-based wrapper for feature selection: An implementation on financial distress identification with support vector machine," Applied Soft Computing, vol. 19, pp. 57–67, 2014.

[44] L. Ma, M. Li, Y. Gao, T. Chen, X. Ma, and L. Qu, "A novel wrapper approach for feature selection in object-based image classification using polygon-based cross-validation," IEEE Geoscience and Remote Sensing Letters, vol. 14, no. 3, pp. 409–413, 2017.

[45] Y. Jiang, X. Liu, G. Yan, and J. Xiao, "Modified binary cuckoo search for feature selection: a hybrid filter-wrapper approach," in 2017 13th International Conference on Computational Intelligence and Security (CIS). IEEE, 2017, pp. 488–491.

[46] K. Javed, H. A. Babri, and M. Saeed, "Feature selection based on class-dependent densities for high-dimensional binary data," IEEE Transactions on Knowledge and Data Engineering, vol. 24, no. 3, pp. 465–477, 2010.

[47] A. Yassine, C. Mohamed, and A. Zinedine, "Feature selection based on pairwise evalution," in 2017 Intelligent Systems and Computer Vision (ISCV). IEEE, 2017, pp. 1–6.

[48] M. M. Mafarja and S. Mirjalili, "Hybrid whale optimization algorithm with simulated annealing for feature selection," Neurocomputing, vol. 260, pp. 302–312, 2017.

[49] Z. Hu, Y. Bao, T. Xiong, and R. Chiong, "Hybrid filter–wrapper feature selection for short-term load forecasting," Engineering Applications of Artificial Intelligence, vol. 40, pp. 17–27, 2015.

[50] B. Agarwal and N. Mittal, "Sentiment classification using rough set based hybrid feature selection," in Proceedings of the 4th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis, 2013, pp. 115–119.

[51] P. Moradi and M. Gholampour, "A hybrid particle swarm optimization for feature subset selection by integrating a novel local search strategy," Applied Soft Computing, vol. 43, pp. 117–130, 2016.

[52] B. Zheng, S. W. Yoon, and S. S. Lam, "Breast cancer diagnosis based on feature extraction using a hybrid of k-means and support vector machine algorithms," Expert Systems with Applications, vol. 41, no. 4, pp. 1476–1482, 2014.

[53] G. Jothi et al., "Hybrid tolerance rough set–firefly based supervised feature selection for mri brain tumor image classification," Applied Soft Computing, vol. 46, pp. 639–651, 2016.

[54] H. H. Inbarani, A. T. Azar, and G. Jothi, "Supervised hybrid feature selection based on pso and rough sets for medical diagnosis," Computer methods and programs in biomedicine, vol. 113, no. 1, pp. 175–185, 2014.

[55] D. Tomar and S. Agarwal, "Hybrid feature selection based weighted least squares twin support vector machine approach for diagnosing breast cancer, hepatitis, and diabetes," Advances in Artificial Neural Systems, vol. 2015, p. 1, 2015.

[56] T. G. S., S. R. Joshi, S. S. Iyengar, N. R. Sunitha, and P. Badrinath, "Mini-batch normalized mutual information: A hybrid feature selection method," IEEE Access, vol. 7, pp. 116 875–116 885, 2019.

[57] T. G. S., D. Jimenez, S. S. Iyengar, J. Miller, N. R. Sunitha, and P. Badrinath, "COMB: A hybrid method for cross-validated feature selection," in Proceedings of the 2020 ACM Southeast Conference, ser. ACM SE '20. New York, NY, USA: Association for Computing Machinery, 2020, p. 100–106. [Online]. Available: https://doi.org/10.1145/3374135.3385285

[58] O. Faker and E. Dogdu, "Intrusion detection using big data and deep learning techniques," in Proceedings of the 2019 ACM Southeast Conference. ACM, 2019, pp. 86–93.

[59] X. Zhang, Q. Zhang, M. Chen, Y. Sun, X. Qin, and H. Li, "A two-stage feature selection and intelligent fault diagnosis method for rotating

machinery using hybrid filter and wrapper method," Neurocomputing, vol. 275, pp. 2426–2439, 2018.
[60] Y. Wang and L. Feng, "Hybrid feature selection using component co-occurrence based feature relevance measurement," Expert Systems with Applications, vol. 102, pp. 83–99, 2018.
[61] k-means clustering. [Online]. Available: https://brilliant.org/wiki/k-means-clustering/#citation-5v
[62] A. Rosenberg and J. Hirschberg, "V-measure: A conditional entropy-based external cluster evaluation measure," in Proceedings of the 2007 joint conference on empirical methods in natural language processing and computational natural language learning (EMNLP-CoNLL), 2007, pp. 410–420.
[63] "learn." [Online]. Available: https://scikit-learn.org/stable/
[64] T. D. Detective, "Why we use an 80/20 split for training and test data plus an alternative method," Jan 2020. [Online]. Available: https://towardsdatascience.com/finally-why-we-use-an-80-20-split-for-training-and-test-data-plus-an.alternative-method-oh-yes-edc77e96295d
[65] "What should be the ratio of train test split." [Online]. Available: https://www.kaggle.com/general/174913
[66] "Cloudstor is powered by aarnet." [Online]. Available: https://cloudstor.aarnet.edu.au/plus/index.php/s/2DhnLGDdEECo4ys?path=/UNSW-NB15-CSVFiles
[67] "Click-through rate prediction." [Online]. Available: https://www.kaggle.com/c/avazu-ctr-prediction/data
[68] "Display advertising challenge." [Online]. Available: https://www.kaggle.com/c/criteo-display-ad-challenge/data
[69] "Talkingdata adtracking fraud detection challenge." [Online]. Available: https://www.kaggle.com/c/talkingdata-adtracking-fraud-detection
[70] D. Dua and C. Graff, "UCI machine learning repository," 2017. [Online]. Available: http://archive.ics.uci.edu/ml
[71] O. L. Mangasarian and W. H. Wolberg, "Cancer diagnosis via linear programming," University of Wisconsin-Madison Department of Computer Sciences, Tech. Rep., 1990.
[72] S. Bionetworks, "Sage bionetworks." [Online]. Available: https://www.synapse.org/#!Synapse:syn4301332
[73] T. G. S., K. G. Boroojeni, K. Chandna, I. Bhatia, S. Iyengar, and N. Sunitha, "Deep learning-based model to fight against ad click fraud," in Proceedings of the 2019 ACM Southeast Conference, 2019, pp. 176–181.
[74] T. G. S., J. Soni, K. G. Boroojeni, S. S. Iyengar, K. Srivastava, P. Badrinath, N. R. Sunitha, N. Prabakar, and H. Upadhyay, "A multi-time-scale time series analysis for click fraud forecasting using binary labeled imbalanced dataset," in 2019 4th International Conference on Computational Systems and Information Technology for Sustainable Solution (CSITSS), vol. 4, 2019, pp. 1–8.
[75] "Accuracy, precision, recall & f1 score: Interpretation of performance measures," Nov 2016. [Online]. Available: https://blog.exsilio.com/all/accuracy-precision-recall-f1-score-interpretation-of-performance/
[76] NCSS, "Roccurve." [Online]. Available: OneROCCurveandCutoffAnalysis
[77] H. Jeon and S. Oh, "Hybrid-recursive feature elimination for efficient feature selection," Applied Sciences, vol. 10, no. 9, p. 3211, 2020.
[78] P. M. Granitto, C. Furlanello, F. Biasioli, and F. Gasperi, "Recursive feature elimination with random forest for ptr-ms analysis of agroindustrial products," Chemometrics and Intelligent Laboratory Systems, vol. 83, no. 2, pp. 83–90, 2006.
[79] K. Yan and D. Zhang, "Feature selection and analysis on correlated gas sensor data with recursive feature elimination," Sensors and Actuators B: Chemical, vol. 212, pp. 353–363, 2015.
[80] J. Brownlee, "Recursive feature elimination (rfe) for feature selection in python," Aug 2020. [Online]. Available: https://machinelearningmastery.com/rfe-feature-selection-in-python/
[81] C. Chu, A.-L. Hsu, K.-H. Chou, P. Bandettini, C. Lin, A. D. N. Initiative et al., "Does feature selection improve classification accuracy? impact of sample size and feature selection on classification using anatomical magnetic resonance images," Neuroimage, vol. 60, no. 1, pp. 59–70, 2012.
[82] R. Primartha and B. A. Tama, "Anomaly detection using random forest: A performance revisited," 2017 International Conference on Data and Software Engineering (ICoDSE), 2017.
[83] N. Moustafa and J. Slay, "The significant features of the unsw-nb15 and the kdd99 data sets for network intrusion detection systems," 2015 4th International Workshop on Building Analysis Datasets and Gathering Experience Returns for Security (BADGERS), 2015.
[84] M. Belouch, S. El, and M. Idhammad, "A two-stage classifier approach using reptree algorithm for network intrusion detection," International Journal of Advanced Computer Science and Applications, vol. 8, no. 6, 2017.
[85] M. Belouch, S. E. Hadaj, and M. Idhammad, "Performance evaluation of intrusion detection based on machine learning using apache spark," Procedia Computer Science, vol. 127, p. 1–6, 2018.
[86] M. Al-Zewairi, S. Almajali, and A. Awajan, "Experimental evaluation of a multi-layer feed-forward artificial neural network classifier for network intrusion detection system," 2017 International Conference on New Trends in Computing Sciences (ICTCS), 2017.
[87] M. Liu and D. Zhang, "Pairwise constraint-guided sparse learning for feature selection," IEEE Transactions on Cybernetics, vol. 46, no. 1, p. 298–310, 2016.
[88] M. Ghaemi and M.-R. Feizi-Derakhshi, "Feature selection using forest optimization algorithm," Pattern Recognition, vol. 60, p. 121–129, 2016.
[89] X. Qiu, Y. Zuo, and G. Liu, "Etcf: An ensemble model for ctr prediction," 2018 15th International Conference on Service Systems and Service Management (ICSSSM), 2018.
[90] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," The Journal of Machine Learning Research, vol. 7, pp. 1–30, 2006.
[91] V. López, A. Fernández, S. García, V. Palade, and F. Herrera, "An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics," Information sciences, vol. 250, pp. 113–141, 2013.
[92] E. R. DeLong, D. M. DeLong, and D. L. Clarke-Pearson, "Comparing the areas under two or more correlated receiver operating characteristic curves: a nonparametric approach," Biometrics, pp. 837–845, 1988.
[93] M. P. Fay, "Two-sided exact tests and matching confidence intervals for discrete data," R journal, vol. 2, no. 1, pp. 53–58, 2010.
[94] Q. McNemar, "Note on the sampling error of the difference between correlated proportions or percentages," Psychometrika, vol. 12, no. 2, pp. 153–157, 1947.
[95] M. Friedman, "A comparison of alternative tests of significance for the problem of m rankings," The Annals of Mathematical Statistics, vol. 11, no. 1, pp. 86–92, 1940.
[96] M. Hollander and D. A. Wolfe, Nonparametric Statistical Methods. John Wiley amp; Sons, 1973, p. 115–120.
[97] "Friedman rank sum test." [Online]. Available: https://www.rdocumentation.org/packages/stats/versions/3.6.2/topics/friedman.test

DR. THEJAS G.S. is an Assistant Professor at the Dept. of Computer Science and Electrical Engineering at Tarleton State University (TSU), Texas A&M University System. He recieveed his Ph.D. from the School of Computing and Information Sciences (SCIS), Florida International University (FIU) under the supervision of Dr. S.S. Iyengar and external guidance of Dr. N. R. Sunitha. He received his Bachelor's and Master's degree in Computer Science and Engineering (M.Tech) from Sri Siddhartha Institute of Technology (SSIT). He worked as a trainee at Defence Research and Development Organization/Electronic and RADAR Development Establishment (DRDO/LRDE). His areas of research include Machine Learning, Deep Learning, Cybersecurity, Human Computer Interaction (HCI), and Performance Optimization using parallel computing. Thejas is a recipient of Summer Research Cohort-II award, 2021 Presidential Excellence in research Scholar award at TSU, Dissertation Year Fellowship Award and Best Graduate Student in Service Award at FIU. Thejas worked as an Assistant Professor for five years at Siddaganga Institute of Technology (SIT). His research has successfully produced several publications in top journals and conferences like ACM, IEEE, Springer, MDPI etc.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3112169, IEEE Access

RAMESHWAR GARG graduated from RV College of Engineering, with a degree in the field of Computer Science. He completed his schooling from Bangalore and showed an early interest in Engineering. He developed a passion for conducting research in the field of Machine Learning with a research internship under Thejas G.S. and S.S. Iyengar in FIU, Miami. He was also a research intern at Samsung Research Institute - Bangalore, in the field of Computer Vision & Deep Learning. Having worked on a Machine Learning project with Nokia, his interest in research kept increasing and he is currently working as a Software Engineer in Cisco Systems India, on a project based on Time Series Analysis. He has also been a part of multiple NGOs, focusing on the education and housing of the underprivileged. He continues to show a keen interest in Machine Learning, Deep Learning and Data Analytics.

DR. S.S. IYENGAR is a Leading Researcher in the fields of distributed sensor networks, computational robotics, and oceanographic applications, and is perhaps best known for introducing novel data structures and algorithmic techniques for large scale computations in sensor technologies and image processing applications. He has published over 500 research papers and has authored or co-authored 12 textbooks and edited 10 others. He is a member of the European Academy of Sciences, a fellow of the National Academy of Inventors, a fellow of the Association of Computing Machinery, a fellow of the American Association for the Advancement of Science, and a fellow of the Society for Design and Process Science. He has received the Distinguished Alumnus Award of the Indian Institute of Science. In 1998, he was awarded the IEEE Computer Society's Technical Achievement Award and is an IEEE Golden Core Member. He is an IEEE Distinguished Visitor, a SIAM Distinguished Lecturer, and an ACM National Lecturer. In 2006, his paper entitled A Fast Parallel Thinning Algorithm for the Binary Image Skeletonization, was the most frequently read article in the month of January in the International Journal of High Performance Computing Applications. His innovative work called the Brooks-Iyengar Algorithm along with the Prof. R. Brooks from Clemson University is applied in industries and some real-world applications which has led to get IEEE Test of Time Award in 2019.

DR. N.R. SUNITHA is a Professor at the dept. of CS&E at Siddaganga Institute of Technology, India. She received her PhD from Visveswaraiah Technological University, Belguam, and her MS is from Birla Institute of Technology and Science and her Bachelor of Engineering is from Gulbarga University. Her research interest includes Cryptography & Network Security, Storage Area Networks, Big Data processing, Industrial Automation, and Computer Security and Reliability. She has funded for her research projects from ABB GISL, AICTE, DRDO, IISc, ICT Skill Development Society and so on in India. She has acquired totally 6 patentes in her research field. She is a reviewer for the journals such as Elsevier's Computers and Security and Intl. Journal of Network Security (IJNS). She was a Chairperson in the International conferences CISTM 2007 (Conference on Information Sci. Tech. & Mgt.) CSNA 2010 (Conference on Network Security & Applications) , NCACA, National Conference on Advances in Computer Applications, and International Conference on Advances in Computing, ICAdC 2012. Her biodata included in Marquis Who's is Who in Science & Engineering 2010. She has got IBM Mentor Award during 2014. She has published more than 65 peer reviewed research articles in leading conferences and journals like ACM, Springer, IEEE, Elsevier and others. She possess membership of personal bodies in Association of Computing Machinery, USA (ACM), Indian Society for Technical Education, India (ISTE) – Life Member, Computer Society of India (CSI), IAENG (International Association of Engineers), IEEE, and Institution of Engineers (FIE).

PRAJWAL BADRINATH received his BS degree from REVA University, Bangalore, India in 2017 and is currently working towards his MS degree at Florida International University, Miami, FL, USA. His research interests include Machine Learning and Time Series Analysis.

SHASANK CHENNUPATI is an Research Programmer currently working with the University of North Carolina, Chapel Hill, in the UNC School of Medicine. Shasank received his Masters from School of Public Health, Florida International University (FIU). His areas of expertise include Statistical Analysis, Data Wrangling and Machine Learning. Shasank previously worked as a Statistical programmer with Fred Hutch in Seattle, WA where he was part of successfully published research in Journals like Leukemia and Lymphoma, Blood, and Journal of Thoracic oncology etc.