COMB: A Hybrid Method for Cross-validated Feature Selection

Thejas G. S.
Tarleton State University, Member of The Texas A&M University System
Stephenville, Texas, USA
sadashiva@tarleton.edu

Daniel Jimenez
Florida International University
Miami, Florida, USA
djime072@fiu.edu

S.S. Iyengar
Florida International University
Miami, Florida, USA
iyengar@cis.fiu.edu

Jerry Miller
Florida International University
Miami, Florida, USA
millej@fiu.edu

N.R. Sunitha
Siddaganga Institute of Technology
Tumakuru, Karnataka, India
nrsunitha@sit.ac.in

Prajwal Badrinath
Florida International University
Miami, Florida, USA
pbadr002@fiu.edu

ABSTRACT

When seeking to obtain insights from massive amounts of data, supervised classification problems require preprocessing to optimize computation. Among the various steps in preprocessing, feature selection (FS) empowers machine learning methods only to receive relevant data. We propose hybrid FS methods using unsupervised classification, statistical scoring, and a wrapper method. Among our tests using twelve dataset problems, the increase in performance from our novel method against existing FS methods represents an advancement in supervised classification.

CCS CONCEPTS

• **Computing methodologies** → **Feature selection**; *Supervised learning by classification.*

KEYWORDS

Feature selection, Filter Method, Wrapper Method, Hybrid Feature Selection, Mini-batch K-means, Random Forest, Supervised Machine Learning, Classification, Cross-validation

ACM Reference Format:
Thejas G. S., Daniel Jimenez, S.S. Iyengar, Jerry Miller, N.R. Sunitha, and Prajwal Badrinath. 2020. COMB: A Hybrid Method for Cross-validated Feature Selection. In *2020 ACM Southeast Conference (ACMSE 2020), April 2–4, 2020, Tampa, FL, USA.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3374135.3385285

1 INTRODUCTION

In the past decade, more digital and accessible data has been produced than ever before in human history [55]. Machine learning is being used to enhance the usefulness of large amounts of structured data across business, industry, academia, and recreational pursuits. Applications across these diverse areas have been opening windows for analysts to find correlations between variables which may change processes in these areas as well as improving our lives [42].

One of the critical challenges in classification lies in FS, which is to provide only the most pertinent and least redundant data to the algorithm. When calculating the size of data, the cardinality of samples is multiplied by the cardinality of features. If there are irrelevant features, it unnecessarily multiplies the data needed in processing. When the amount of data is unnecessarily multiplied, the computation time and memory usage may exponentially increase while running classification algorithms. Irrelevant data decreases accurate prediction and performance. When time and memory are increased for training the model, the financial burden of utilizing machine learning will hinder innovation in useful applications. The optimal result of applying FS is an increase in classification performance.

To address the above-said problem, we considered three existing hybrid FS approaches and developed a hybrid FS method. Our approach is called Clustering Optimization with Mini-batch K-means and Cross-validation (COMB).

In developing a new hybrid FS method, we have considered standard FS methods such as Homogeneity and Recursive Feature Elimination (HRFE), Random Forests Feature with Recursive Feature Elimination (RFRFE), and Scalar Vector Machine with Recursive Feature Elimination (SVMRFE), a control case without FS, and are compared with our hybrid solution. All the proposed and standard FS methods have been tested using twelve test cases. Our evaluation metrics were accuracy, precision, F1-score, ROC-AUC, and cardinality of features. During experimentation, the COMB hybrid method has consistently performed well compared to other FS approaches.

Organization of the paper: Section 2 presents various related works of the standard methods. In seciton 3, we present background information of our method. In section 4, we explain our experiment methodology. After we discuss our results in section 5, we share our conclusion in section 6.

2 RELATED WORK

Researchers have explored three principal approaches for feature selection, namely the filter, the wrapper, or a hybrid approach.

Filter Approach. Filter methods are FS algorithms with statistical evaluation criteria. A filter method scores each feature in a dataset. After scoring, an algorithm tests these scores against the desired criteria to determine which features are kept or discarded.

ACMSE 2020, April 2–4, 2020, Tampa, FL, USA

ACM ISBN 978-1-4503-7105-6/20/03…$15.00
https://doi.org/10.1145/3374135.3385285

The threshold score usually determines the feature inclusion decision. Pearson's correlation coefficient [18], mutual information [5], chi-squared [19], linear discriminant analysis [32], linear regression [8], and logistical regression [47] are some statistical methods used as criteria for FS. From statistical methods, researchers derived FS methods such as the following: Joint Mutual Information (JMI)[54], Fast Cluster-Based FS Algorithm (FAST)[49], Minimal Redundancy Maximal Relevance (MRMR) [37], and Double Input Symmetrical Relevance (DISR) [27]. Even though the filter approach may be quicker than other approaches, these approaches do not verify their results with a classification algorithm to improve their selected features.

Wrapper Approach. Unlike the filter method, which only scores each feature against the class label using a statistical method, a wrapper method tests the accuracy of a classification model to determine whether to keep or discard a feature in the subset for the final model. When it meets the stopping criteria, the method will have selected the final features. Early variants of wrapper methods tested the features using a naive Bayes approach [22]. Because of its efficient time complexity, novel FS methods are still incorporating naive Bayes such as Naive Feature Selection (NFS) [4]. Some methods which utilize other classification methods are the Cosine Similarity Measure Support Vector Machines (CSMSVM) [11], a Genetic Algorithm and Logistical Regression (GA-LR) [20], and Local Maximization and Floating maximization (LM-FM) [23]. In 2015, Panthong and Srivihok proposed four new methods Evolutionary selection and Bagging and Naive Bayes (OBDT), Evolutionary selection and Bagging and Naive Bayes (OBNB), Evolutionary selection and AdaBoost and Decision Tree (OADT), and Evolutionary selection and AdaBoost and Naive Bayes (OANB) [33]. Their study tests an evolutionarily optimized selection and utilizes a heuristic search for FS. The optimized selection preceded bagging [9] or Ada boosting [15] with naive Bayes or decision trees to evaluate the feature subset. In Panthong and Srivihok's experiments, the gain in accuracy only improved upon the results of existing FS methods in datasets with small dimensionality, which were only 3 out of the 14 datasets. There were less than ten features in each of those three datasets. There are also recently implemented methods such as RFRFE, and SVMRFE that use recursively eliminates the lowest-ranked features of the random forest or SVM classifier respectively [45] [56]. Wrapper methods may involve combinations of classification methods to improve accuracy, yet they may also increase the average running time complexity for datasets with a large number of features.

Hybrid Approach. A hybrid method combines both a filter method and a wrapper method as part of the FS process. Like filter methods, statistical methods provide relational information about each sample's features with its classification label. As a wrapper method does, the hybrid method also uses classification to test the validity of the entirety of the features in simulated models [17]. Some hybrid methods implement evolutionary approaches such as Cellular Learning Automata with Ant Colony Optimization [46], Binary Black Hole Algorithm (BBHA) with RF ranking [35], Biography Algorithm with Support Vector Machine (SVM) [24], and Harmony Search [48]. Venkatesh and Anuradha implemented a hybrid method that combines the wrapper method RFRFE with a mutual information filter [53]. One of the latest hybrid methods used for FS on large datasets while proving high accuracy is Homogeneity and Recursive Feature Elimination (HRFE) [13]. For each feature, HRFE uses k-means to cluster each sample with a unique sample label. The homogeneity score is considered between the predicted labels for all samples using one feature and the actual labels. Each feature is then sorted in descending order before beginning recursive feature elimination. The subset with the highest random forest accuracy are the selected features. Other hybrid FS algorithms include Correlation-Based Hybrid Feature Selection (CBHFS) [34], random forests feature importance [21], and Mini-Batch Normalized Mutual Information (KNFI & KNFE) [41]. The existing hybrid method limitation is that there is no balanced cross-validation done while testing the classification accuracy.

3 PROPOSED APPROACH

3.1 Preliminaries

Mini-batch K-Means. K-means groups data points into y clusters. First, it places y centroids randomly among the dataset. For every data point, the algorithm takes the Euclidean distance for each of they centroids. The calculation of the Euclidean distance is mathematically represented as shown below in equation 1.

$$Y(y,z) = \sqrt{(w_1 - w_2)^2 + (x_1 - x_2)^2} \tag{1}$$

where w and x are the coordinates of a cluster centroid, y, and a data point, z [30]. The algorithm clusters the data point with the nearest centroid. The mean of each cluster's data points is the respective centroid's updated coordinates. The algorithm terminates when the adjustment of centroids stops changing [3]. Since the mini-batch k-means variant uses subsets from the input, it has a quicker CPU runtime than k-means for larger datasets [2]. Therefore, some applications display mini-batch k-means' advantage over k-means [14].

Completeness & Homogeneity. Completeness provides a score to test if all class members are within a given cluster, and homogeneity provides a high score when each cluster only contains data points, which are all of the same class [39]. The conditional entropy of the clusters determines the completeness score given by the class assignment divided by the entropy of the cluster. If the entropy increases, the element of surprise increases. Completeness is symmetrically related to homogeneity. Completeness may be represented as shown in equation 2:

$$COM = 1 - \frac{E(H,C)}{E(H)} \tag{2}$$

$$HOM = 1 - \frac{E(C,H)}{E(C)} \tag{3}$$

$$E(H,C) = -\sum_{c=1}^{C}\sum_{h=1}^{H}\frac{n_{c,h}}{n}\log\frac{n_{c,h}}{n_c} \tag{4}$$

$$E(H) = -\sum_{h=1}^{H}\frac{n_h}{n}\log\frac{n_h}{n} \tag{5}$$

where COM is completeness, HOM is homogeneity in equation 3, $E(H,C)$ is the conditional entropy in equation 4 and in equation 5 $E(H)$ is the entropy of a cluster that contains n samples, $n_{c,h}$ is the cardinality of samples from cluster h allocated to class c, and n_c and n_h are the cardinality of samples belonging to class c and cluster h

respectively. Given a classification model $\zeta(x) = y$, x is the sample with $features[f1...fn]$ and y is the output label. If the completeness score is closer to 1, the predicted y value entropy is closer to the entropy of the actual y values. Therefore, the classification can rely upon the feature which was used for the clustering process. A lower score shows that the attribute is less significant in classification.

Algorithm 1: COMB

Input: X ← Feature Set with n features $f_0, f_1, ...f_{n-1}$
Y ← Labels for all samples

```
score ← [ ]
for i ← 0 to n − 1 do
    K ← countUnique( Y )
    P ← mKmeans( X[f_i] , K )
    score[f_i] ← meanOfHCV( Y , P )
end
X' ← descending( score , X )
max ← 0
for i ← 0 to n − 1 do
    new ← concat( new , X'[i] )
    cv ← [ ]
    for train, test in S5FoldCV( new , Y ) do
        prediction ← RandomForest( X'[train] , Y[train] )
        cv ← concat( cv , prediction )
    end
    if mean(cv) >= max then
        max ← mean( cv )
    else
        new ← removeLastColumn( new )
    end
end
```

Output: X ← new

V-Measure. Given the homogeneity score and the completeness score, one may obtain the v-measure. The V-measure is the harmonic mean of completeness and homogeneity. The mathematical representation is [39] in equation 6:

$$V = \frac{\theta \cdot \psi}{\theta + \psi} \cdot 2 \tag{6}$$

where θ and ψ refer to completeness and homogeneity respectively.

Random Forests. The most commonly used methods for decision trees are random forests [7]. During classification, the random forest ensemble learning method creates multiple trees. The class with the majority vote is the sample's predicted class [25]. Random forest scales for large datasets with many samples and features [26]. Since each tree is trained independent of the other, RF may take advantage of multiprocessing to improve it's running time. Because of it's speed, realtime face tracking applications have also utilized it [52]. We have chosen to use this classifier in our method because of it's ability to scale with large datasets and utilize multiprocessing to improve speed.

3.2 Our Approach

The proposed FS method's framework is as follows:

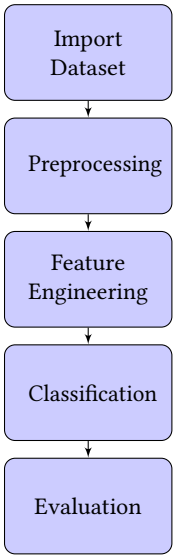

Figure 1: Experiment Process

(1) For each feature, we predict classes for each sample using a mini-batch k-means with X clusters. X is the number of unique classes.
(2) We rank each feature using the average of v-measure, completeness, and homogeneity of the predicted classes with the actual classes.
(3) We then use feature addition to verify whether adding the next highest feature improves the accuracy.
(4) The method returns the subset, which presents the highest accuracy with the stratified k-fold cross-validated random forest model.

By using feature addition, our method should perform at a better running time complexity than the compared methods in the best case, which is when there is only one relevant feature in classification. In recursive feature elimination, the worst case would be $O((d * n)^2)$ since RFE begins with the full set of features. With feature addition, the best case would be $O(d * n)$ since feature addition begins with one feature. This is the case where the number of features is d, and the number of samples is n. Algorithm 1 shows our hybrid FS COMB and Figure 1 depicts the experimental process of the work.

4 EXPERIMENT

4.1 Datasets

We are experimenting with the following eleven datasets to test our approach of FS. Criteo [1], Avazu [2], and Talking [3], and Sonar [4] are large datasets which have been used in Kaggle [5] competitions. The University of California in Irvine machine learning repository [12] hosts the rest of the datasets used in the experiments.

[1] https://www.kaggle.com/c/criteo-display-ad-challenge/data
[2] https://www.kaggle.com/c/avazu-ctr-prediction/data
[3] https://www.kaggle.com/c/talkingdata-adtracking-fraud-detection/data
[4] https://www.kaggle.com/c/sonar-data-set/data
[5] https://www.kaggle.com/kaggle

The entire UNSW-NB15 created by Cyber Range Lab of the Australian Center for Cyber Security (ACCS) is a dataset that contains two million network security samples [28]. We will experiment with this dataset for both binary and multiclass classification. Table 1-2 shows the description of the datasets.

Table 1: Datasets for Binary Classification

Name	Samples	Features
Avazu	1,000,000	27
Breast Cancer	699	10
Ionoshpere	351	34
Spambase	4,601	57
Sonar	209	60
Talking	1,000,000	11
UNSW-NB15	2,540,047	43

Table 2: Datasets for Multiclass Classification

Name	Samples	Features	Labels
Abalone	4,771	8	3
Gene	801	20,531	5
Iris	150	4	4
Lung Cancer	32	56	3
UNSW-NB15	2,540,047	43	10

4.2 Preprocessing

Preprocessing removes anomalies that would render data processing difficult. All datasets will have their uninterpretable values imputed with the most common value to prevent type errors. Since the binary classification datasets contain a higher number of samples, the datasets for multiclass classification incorporates a data normalization technique called standard scalar [16]. Standard scalar transforms the values of each feature into a normal distribution to prevent the classification algorithm from compensating for a high variance of values. The equation for the imputed value is defined as in 7:

$$S(\epsilon) = \frac{\epsilon - \bar{\epsilon}}{\sigma} \tag{7}$$

where ϵ is the sample's feature value, $\bar{\epsilon}$ is mean of the feature's values, and σ is the feature's standard deviation.

According to S. et al., the Talking dataset has been reduced to a million samples and maintains the original dataset class sample ratio [40, 43, 44].

Since the UNSW-NB15 dataset will be used for both binary and multiclass classification experiments, there will be extra steps in the preprocessing. Before data normalization, samples with null values were dropped because they were outliers. In preprocessing, the socket information and IP address columns will be removed to prevent an overfitted model. In order to identify the type of intrusion for multiclass classification, the experimentation will only include samples that are intrusion positive. For binary classification, each sample will be labeled either intrusion positive or intrusion negative.

Table 3: Previous Work's Accuracy on the Binary UNSW-NB15 Dataset

Study	Method	Accuracy
Primartha and Tama [38]	Random Forest	95.5%
Moustafa and Slay [29]	Naive Bayes	79.50%
	Linear Regression	83.00%
	Expectation-Maximization	77.20%
Belouch et al. [6]	Random Tree	86.59%
	Naive Bayes	80.40%
	RepTree	87.80%
	Artificial Neural Network	86.31%
	Decision Tree	86.13%
Al-Zewairi et al. [1]	Deep Learning	98.99%
Faker and Dogdu [13]	Gradient Boosted Tree	97.92%
	Random Forest	98.86%
	Deep Neural Network	99.19%
Our Work	COMB	**99.79%**

Table 4: Previous Work's Accuracy on the Multiclass UNSW-NB15 Dataset

Study	Method	Accuracy
Belouch et al. [6]	Random Tree	76.21%
	Naive Bayes	73.86%
	RepTree	79.20%
	Decision Tree	78.73%
	Artificial Neural Network	78.14%
Our Work	COMB	**89.95%**

In many most datasets, the number of samples among classes is not equal. This a problem for some classification methods such as support vector machines, backpropagation neural networks, and Bayesian networks [51]. Therefore any dataset used for classification should be balanced for optimal results. Decision tree classification methods such as random forests have a bias towards the class with the majority of samples [50]. If there is a class with fewer samples, the classification algorithm will often classify those samples incorrectly. A standard solution to this problem is the Synthetic Minority Oversampling Technique (SMOTE) [10]. To increase the samples of the minority class, SMOTE takes the k-nearest neighbors of each sample using euclidean distance and takes the number of nearest neighbors needed to balance the data. Finally, the new samples are then generated using a sample and its k-nearest neighbors.

4.3 Experimental Setup

All of the experiments were performed in Python language using scikit-learn library [36]. The experiments on the smaller datasets were conducted on Google Collaboratory[6]. For experiments on

[6]https://research.google.com/colaboratory/faq.html

Table 5: Our Results on Multiclass Datasets

Type	Dataset	Method	Accuracy	Precision	Recall	F1	ROC_AUC	Features
Multiclass	Abalone	COMB	57.51%	57.07%	57.51%	56.60%	68.13%	7
		HRFE	56.24%	55.70%	56.24%	55.24%	67.18%	8
		RFRFE	56.24%	55.70%	56.24%	55.24%	67.18%	8
		None	56.24%	55.70%	56.24%	55.24%	67.18%	8
		SVMRFE	55.78%	55.25%	55.78%	55.04%	66.84%	6
	Gene	COMB	99.80%	99.81%	99.80%	99.80%	99.88%	19
		RFRFE	99.80%	99.81%	99.80%	99.80%	99.88%	4262
		None	99.67%	99.68%	99.67%	99.67%	99.79%	20531
		HRFE	99.40%	99.41%	99.40%	99.40%	99.63%	20226
	Iris	COMB	97.33%	97.58%	97.33%	97.33%	98.00%	2
		None	96.00%	96.34%	96.00%	95.98%	97.00%	4
		SVMRFE	94.67%	95.18%	94.67%	94.62%	96.00%	2
		RFRFE	94.67%	95.18%	94.67%	94.62%	96.00%	2
		HRFE	94.00%	94.34%	94.00%	93.96%	95.50%	3
	Lung Cancer	COMB	90.00%	93.44%	90.00%	89.16%	92.50%	4
		HRFE	76.67%	83.56%	76.67%	74.32%	82.50%	9
		RFRFE	76.67%	77.33%	76.67%	74.43%	82.50%	9
		SVMRFE	76.67%	75.11%	76.67%	76.67%	82.50%	13
		None	58.89%	57.89%	58.89%	54.56%	69.17%	56
	UNSW-NB15	COMB	89.95%	90.32%	89.95%	88.69%	94.03%	14
		RFRFE	89.15%	88.94%	89.15%	88.30%	93.33%	6
		HRFE	86.25%	85.50%	86.25%	85.45%	89.41%	17
		None	83.74%	84.06%	83.74%	83.04%	85.72%	43

more massive datasets, we used Florida International University's "Flounder" server with a 64 core AMD Opteron Processor and 504GB of RAM. GitHub link to our work is available in the footnote[7]

For each dataset, we analyze performance with stratified 5-fold cross-validation on the random forest classification method with ten trees in the forest. Random forest is an ensemble method commonly used in classification [31]. The balance of samples for each class is preserved for every fold. We evaluated accuracy, precision, recall, F1-score, the cardinality of features, and ROC-AUC.

5 RESULTS AND DISCUSSION

In the gene dataset, COMB significantly reduced the number of features for the selected feature subset while maintaining an increase of accuracy. As mentioned in section 3, our method uses feature addition. Using feature addition instead of recursive feature elimination, COMB significantly reduced the average number of features in each iteration of cross-validation in the wrapper section of our hybrid method. Because of this, we may expect the highest accuracy and optimized speed of classification using COMB's selected features.

In the Avazu and talking dataset, the control case and HRFE have the same features, yet they have different accuracy. This may be attributed to RF and the number of trees. In each tree, a subset of columns in the dataset is selected at random. Before HRFE begins recursive feature elimination, it sorts the columns in descending order by homogeneity score. Since the columns are rearranged and have different indexes, the classifier's trees are will not always select the same columns; the final performance analysis accuracy varies between the methods. In future work, we hypothesize that the optimal number of trees in a random forest is found when the columns can be rearranged, and the accuracy stays constant.

In the breast cancer dataset, the control case held the highest accuracy, yet COMB maintained the metrics of the control case and reduced it's features. Since the creators of the dataset have optimized the dataset with ten features, there are fewer opportunities to optimize classification with feature selection methods. The experiment for the breast cancer dataset shows that COMB may maintain the highest performance in datasets with little room for optimization and few features.

In all of datasets, as shown in Table 5-6, COMB performed the highest because of its hybrid approach by combining filter and wrapper capabilities. It improves upon the wrapper by utilizing stratified five fold cross-validation with a feature addition. The filter method also is improved by taking the average of clustering metrics after mini-batch k-means clustering.

As shown in Table 3 and 4, our method exceeds the accuracy of existing works on the UNSW-NB15 dataset.

6 CONCLUSION

After reviewing the advantages and disadvantages of existing approaches, we present a hybrid method that utilizes clustering for

[7]https://github.com/thejasgs/COMB

Table 6: Our Results on Binary Datasets

Type	Dataset	Method	Accuracy	Precision	Recall	F1	ROC_AUC	Features
Binary	Avazu	COMB	86.85%	89.15%	86.85%	85.83%	86.85%	20
		None	86.63%	88.81%	86.63%	85.70%	86.63%	27
		RFRFE	86.58%	88.75%	86.58%	85.65%	86.58%	23
		HRFE	86.55%	88.77%	86.55%	85.59%	86.55%	27
	Breast Cancer	None	97.38%	97.40%	97.38%	97.38%	97.38%	10
		COMB	97.38%	97.40%	97.38%	97.38%	97.38%	8
		SVMRFE	97.27%	97.30%	97.27%	97.27%	97.27%	8
		RFRFE	97.06%	97.12%	97.06%	97.06%	97.06%	8
		HRFE	96.62%	96.64%	96.62%	96.62%	96.62%	9
	Ionoshpere	COMB	93.33%	93.60%	93.33%	93.32%	93.33%	10
		None	92.67%	92.81%	92.67%	92.66%	92.67%	34
		RFRFE	92.67%	92.80%	92.67%	92.66%	92.67%	23
		HRFE	92.67%	92.96%	92.67%	92.64%	92.67%	27
		SVMRFE	92.00%	92.16%	92.00%	91.99%	92.00%	23
	Sonar	COMB	79.78%	80.04%	79.78%	79.74%	79.74%	9
		HRFE	75.25%	75.80%	75.25%	75.04%	75.26%	7
		SVMRFE	73.45%	73.66%	73.45%	73.17%	73.44%	59
		RFRFE	68.07%	68.34%	68.07%	67.80%	68.02%	50
		None	66.26%	66.62%	66.26%	65.79%	66.28%	60
	Spambase	COMB	93.24%	93.55%	93.24%	93.22%	93.24%	30
		HRFE	92.95%	93.48%	92.95%	92.90%	92.95%	55
		RFRFE	92.75%	93.36%	92.75%	92.70%	92.76%	37
		SVMRFE	92.31%	92.88%	92.31%	92.25%	92.31%	47
		None	92.16%	92.82%	92.16%	92.10%	92.17%	57
	Talking	COMB	97.91%	97.97%	97.91%	97.91%	97.91%	9
		RFRFE	97.48%	97.55%	97.48%	97.48%	97.48%	6
		None	63.12%	74.00%	63.12%	57.01%	63.12%	11
		SVMRFE	63.12%	74.00%	63.12%	57.01%	63.12%	11
		HRFE	62.82%	73.26%	62.82%	56.20%	62.82%	11
	UNSW-NB15	COMB	99.76%	99.77%	99.76%	99.76%	99.76%	9
		None	93.99%	96.25%	93.99%	93.40%	93.99%	43
		HRFE	93.96%	96.18%	93.96%	93.38%	93.96%	3
		RFRFE	91.15%	95.30%	91.15%	89.00%	91.15%	28

feature ranking and cross-validation. Our method uses the average of completeness, homogeneity, and v-measure metrics to score the mini-batch k-means clustering. Once the method ranks the feature, the method examines whether the next highest-ranked feature improves the classification through feature addition with stratified 5-fold cross-validation. If adding the feature improves classification, the feature is included. This method allows for faster FS on datasets with many features. Following the introduction of how our method works, we tested our method, existing methods, and a control case on 11 datasets for binary and multiclass classification. In the results of our experimentation, COMB selected the least number of features that provide the classifier with the highest prediction accuracy for most of the datasets. In future work, our method may dynamically choose the optimal number of trees in the classifier for higher accuracy and continue to optimize the speed of FS.

REFERENCES

[1] M. Al-Zewairi, S. Almajali, and A. Awajan. 2017. Experimental evaluation of a multi-layer feed-forward artificial neural network classifier for network intrusion detection system. In *2017 International Conference on New Trends in Computing Sciences (ICTCS)*. IEEE, Amman, Jordan, 167–172.

[2] J. Béjar Alonso. 2013. K-means vs Mini Batch K-means: A comparison. (2013).

[3] D. Arthur and S. Vassilvitskii. 2007. k-means++: The advantages of careful seeding. In *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*. Society for Industrial and Applied Mathematics, 1027–1035.

[4] A. Askari, A. dÁspremont, and L. El Ghaoui. 2019. Naive Feature Selection: Sparsity in Naive Bayes. *arXiv preprint arXiv:1905.09884* (2019).

[5] R. Battiti. 1994. Using mutual information for selecting features in supervised neural net learning. *IEEE Transactions on neural networks* 5, 4 (1994), 537–550.

[6] M. Belouch, S. El Hadaj, and M. Idhammad. 2017. A two-stage classifier approach using reptree algorithm for network intrusion detection. *International Journal of Advanced Computer Science and Applications* 8, 6 (2017), 389–394.

[7] G. Biau and E. Scornet. 2016. A random forest guided tour. *Test* 25, 2 (2016), 197–227.

[8] P. Breheny and J. Huang. 2011. Coordinate descent algorithms for nonconvex penalized regression, with applications to biological feature selection. *The annals of applied statistics* 5, 1 (2011), 232.

[9] L. Breiman. 1996. Bagging predictors. *Machine learning* 24, 2 (1996), 123–140.

[10] N. V Chawla, K. W Bowyer, L. O Hall, and W. Philip Kegelmeyer. 2002. SMOTE: synthetic minority over-sampling technique. *Journal of artificial intelligence research* 16 (2002), 321–357.

[11] G. Chen and J. Chen. 2015. A novel wrapper method for feature selection and its applications. *Neurocomputing* 159 (2015), 219–226.

[12] D. Dua and C. Graff. 2017. UCI Machine Learning Repository. http://archive.ics.uci.edu/ml

[13] O. Faker and E. Dogdu. 2019. Intrusion Detection Using Big Data and Deep Learning Techniques. In *Proceedings of the 2019 ACM Southeast Conference (ACM SE '19)*. Association for Computing Machinery, New York, NY, USA, 86–93. https://doi.org/10.1145/3299815.3314439

[14] A. Feizollah, N. Badrul Anuar, R. Salleh, and F. Amalina. 2014. Comparative study of k-means and mini batch k-means clustering algorithms in android malware detection using network traffic analysis. In *2014 International Symposium on Biometrics and Security Technologies (ISBAST)*. IEEE, Kuala Lumpur, Malaysia, 193–197.

[15] Y. Freund and R. E Schapire. 1997. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences* 55, 1 (1997), 119–139.

[16] S. García, S. Ramírez-Gallego, J. Luengo, J. Manuel Benítez, and F. Herrera. 2016. Big data preprocessing: methods and prospects. *Big Data Analytics* 1, 1 (2016), 9.

[17] H. Hsu, C. Hsieh, and M. Lu. 2011. Hybrid feature selection by combining filters and wrappers. *Expert Systems with Applications* 38, 7 (2011), 8144–8150.

[18] B. Jacek and W. Duch. 2007. *Feature Selection for High-Dimensional Data — A Pearson Redundancy Based Filter*. Vol. 45. 242–249. https://doi.org/10.1007/978-3-540-75175-5_30

[19] X. Jin, A. Xu, R. Bie, and P. Guo. 2006. Machine learning techniques and chi-square feature selection for cancer classification using SAGE gene expression profiles. In *International Workshop on Data Mining for Biomedical Applications*. Springer, 106–115.

[20] C. Khammassi and S. Krichen. 2017. A GA-LR wrapper approach for feature selection in network intrusion detection. *computers & security* 70 (2017), 255–277.

[21] D. Seong Kim, S. Min Lee, and J. Sou Park. 2006. Building lightweight intrusion detection system based on random forest. In *International Symposium on Neural Networks*. Springer, 224–230.

[22] R. Kohavi and G. H John. 1997. Wrappers for feature subset selection. *Artificial intelligence* 97, 1-2 (1997), 273–324.

[23] V. Ch Korfiatis, P. A Asvestas, K. K Delibasis, and G. K Matsopoulos. 2013. A classification system based on a new wrapper feature selection algorithm for the diagnosis of primary and secondary polycythemia. *Computers in biology and medicine* 43, 12 (2013), 2118–2126.

[24] X. Li and M. Yin. 2013. Multiobjective binary biogeography based optimization for feature selection using gene expression data. *IEEE Transactions on NanoBioscience* 12, 4 (2013), 343–353.

[25] A. Liaw and M. Wiener. 2002. Classification and regression by randomForest. *R news* 2, 3 (2002), 18–22.

[26] Y. Liu. 2014. Random forest algorithm in big data environment. *Computer Modelling & New Technologies* 18, 12A (2014), 147–151.

[27] P. E. Meyer and G. Bontempi. 2006. On the Use of Variable Complementarity for Feature Selection in Cancer Classification. In *Applications of Evolutionary Computing*. Springer Berlin Heidelberg, Berlin, Heidelberg, 91–102.

[28] N. Moustafa and J. Slay. 2015. UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). In *2015 military communications and information systems conference (MilCIS)*. IEEE, Canberra, ACT, Australia, 1–6.

[29] N. Moustafa and J. Slay. 2017. A hybrid feature selection for network intrusion detection systems: Central points. *arXiv preprint arXiv:1707.05505* (2017).

[30] J. Nicholson and C. Clapham. 2014. *The Concise Oxford Dictionary of Mathematics*. Vol. 5. Oxford University Press Oxford.

[31] T. Mayumi Oshiro, P. Santoro Perez, and J. Augusto Baranauskas. 2012. How many trees in a random forest?. In *International workshop on machine learning and data mining in pattern recognition*. Springer, 154–168.

[32] S. Pang, S. Ozawa, and N. Kasabov. 2005. Incremental linear discriminant analysis for classification of data streams. *IEEE transactions on Systems, Man, and Cybernetics, part B (Cybernetics)* 35, 5 (2005), 905–914.

[33] R. Panthong and A. Srivihok. 2015. Wrapper feature subset selection for dimension reduction based on ensemble learning algorithm. *Procedia Computer Science* 72 (2015), 162–169.

[34] J. Sou Park, K. Mohammad Shazzad, and D. Seong Kim. 2005. Toward Modeling Lightweight Intrusion Detection System Through Correlation-Based Hybrid Feature Selection. In *Information Security and Cryptology*, D. Feng, D. Lin, and M. Yung (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 279–289.

[35] E. Pashaei, M. Ozen, and N. Aydin. 2016. Gene selection and classification approach for microarray data based on Random Forest Ranking and BBHA. In *2016 IEEE-EMBS International Conference on Biomedical and Health Informatics (BHI)*. IEEE, Las Vegas, NV, USA, 308–311.

[36] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, and V. Dubourg. 2011. Scikit-learn: Machine learning in Python. *Journal of machine learning research* 12, Oct (2011), 2825–2830.

[37] H. Peng, F. Long, and C. Ding. 2005. Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on Pattern Analysis & Machine Intelligence* 8 (2005), 1226–1238.

[38] R. Primartha and B. Adhi Tama. 2017. Anomaly detection using random forest: A performance revisited. In *2017 International Conference on Data and Software Engineering (ICoDSE)*. IEEE, Palembang, Indonesia, 1–6.

[39] A. Rosenberg and J. Hirschberg. 2007. 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)*. Association for Computational Linguistics, Prague, Czech Republic, 410–420.

[40] T. G. S., K. G Boroojeni, K. Chandna, I. Bhatia, S. S. Iyengar, and N. R. Sunitha. 2019. Deep Learning-based Model to Fight Against Ad Click Fraud. In *Proceedings of the 2019 ACM Southeast Conference (ACM SE '19)*. Association for Computing Machinery, New York, NY, USA, 176–181. https://doi.org/10.1145/3299815.3314453

[41] T. G. S., S. Raj Joshi, S. S. Iyengar, N. R. Sunitha, and P. Badrinath. 2019. Mini-Batch Normalized Mutual Information: A Hybrid Feature Selection Method. *IEEE Access* 7 (2019), 116875–116885.

[42] T. G. S., T. C. P., S. S. Iyengar, and N. R. Sunitha. 2018. Intelligent Access Control: A Self-Adaptable Trust-Based Access Control (SATBAC) Framework Using Game Theory Strategy. In *Proceedings of International Symposium on Sensor Networks, Systems and Security*. Springer International Publishing, Cham, pp. 97–111. https://doi.org/10.1007/978-3-319-75683-7_7

[43] T. G. S., J. Soni, K. G Boroojeni, S. S. Iyengar, K. Srivastava, P. Badrinath, N. R. Sunitha, N. Prabakar, and H. Upadhyay. 2019. A Multi-time-scale Time Series Analysis for Click Fraud Forecasting using Binary Labeled Imbalanced Dataset. (2019), 1–9.

[44] T. G. S., J. Soni, K. Chandna, S. S. Iyengar, N. R. Sunitha, and N. Prabakar. 2019. Learning-based model to fight against fake like clicks on instagram posts. In *IEEE SoutheastCon*. Alabama, USA, 1–8.

[45] H. Sanz, C. Valim, E. Vegas, J. M Oller, and F. Reverter. 2018. SVM-RFE: selection and visualization of the most relevant features through non-linear kernels. *BMC bioinformatics* 19, 1 (2018), 432.

[46] F. Vafaee Sharbaf, S. Mosafer, and M. Hossein Moattar. 2016. A hybrid gene selection approach for microarray data classification using cellular learning automata and ant colony optimization. *Genomics* 107, 6 (2016), 231–238.

[47] S. Krishnaj Shevade and S. Sathiya Keerthi. 2003. A simple and efficient algorithm for gene selection using sparse logistic regression. *Bioinformatics* 19, 17 (2003), 2246–2253.

[48] S. Salameh Shreem, S. Abdullah, and M. Zakree Ahmad Nazri. 2016. Hybrid feature selection algorithm using symmetrical uncertainty and a harmony search algorithm. *International Journal of Systems Science* 47, 6 (2016), 1312–1329.

[49] Q. Song, J. Ni, and G. Wang. 2011. A fast clustering-based feature subset selection algorithm for high-dimensional data. *IEEE transactions on knowledge and data engineering* 25, 1 (2011), 1–14.

[50] C. Strobl, A. Boulesteix, A. Zeileis, and T. Hothorn. 2007. Bias in random forest variable importance measures: Illustrations, sources and a solution. *BMC bioinformatics* 8, 1 (2007), 25.

[51] Y. Sun, A. KC Wong, and M. S Kamel. 2009. Classification of imbalanced data: A review. *International Journal of Pattern Recognition and Artificial Intelligence* 23, 04 (2009), 687–719.

[52] Y. Tang. 2010. *Real-Time Automatic Face Tracking Using Adaptive Random Forests*. Ph.D. Dissertation. McGill University Library.

[53] B. Venkatesh and J. Anuradha. 2019. A Hybrid Feature Selection Approach for Handling a High-Dimensional Data. In *Innovations in Computer Science and Engineering*. Springer, 365–373.

[54] H. Hua Yang and J. Moody. 1999. Data Visualization and Feature Selection: New Algorithms for Nongaussian Data. In *Proceedings of the 12th International Conference on Neural Information Processing Systems (NIPS'99)*. MIT Press, Cambridge, MA, USA, 687–693.

[55] Y. Zhai, Y. Ong, and I. W Tsang. 2014. The emerging" big dimensionality". (2014).

[56] C. Zhang, Y. Li, Z. Yu, and F. Tian. 2016. Feature selection of power system transient stability assessment based on random forest and recursive feature elimination. In *2016 IEEE PES Asia-Pacific Power and Energy Engineering Conference (APPEEC)*. IEEE, Xian, China, 1264–1268.