

Research Article

Statistical Analysis of the Performance of Rank Fusion Methods Applied to a Homogeneous Ensemble Feature Ranking

Majid Soheili ¹, Amir-Masoud Eftekhari Moghadam ¹, and Mehdi Dehghan²

¹Faculty of Computer and Information Technology Engineering, Qazvin Branch, Islamic Azad University, Qazvin, Iran

²Computer Engineering and Information Technology Department, Amirkabir University of Technology, Tehran, Iran

Correspondence should be addressed to Amir-Masoud Eftekhari Moghadam; eftekhari@qiau.ac.ir

Received 18 July 2020; Accepted 21 August 2020; Published 10 September 2020

Academic Editor: Roberto Natella

Copyright © 2020 Majid Soheili et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The feature ranking as a subcategory of the feature selection is an essential preprocessing technique that ranks all features of a dataset such that many important features denote a lot of information. The ensemble learning has two advantages. First, it has been based on the assumption that combining different model's output can lead to a better outcome than the output of any individual models. Second, scalability is an intrinsic characteristic that is so crucial in coping with a large scale dataset. In this paper, a homogeneous ensemble feature ranking algorithm is considered, and the nine rank fusion methods used in this algorithm are analyzed comparatively. The experimental studies are performed on real six medium datasets, and the area under the feature-forward-addition curve criterion is assessed. Finally, the statistical analysis by repeated-measures analysis of variance results reveals that there is no big difference in the performance of the rank fusion methods applied in a homogeneous ensemble feature ranking; however, this difference is a statistical significance, and the B-Min method has a little better performance.

1. Introduction

During recent years, the amount of data generated daily has grown dramatically. IBM estimated that every day 2.5 Quintillion bytes of data is created, and 90% of the data in the world today has been created in the last two years. Nowadays, such voluminous data are known as big data. The analysis of such massive data on a single machine is impossible or very slow and time-consuming. Hence, it is necessary to use the algorithms which can be distributed between several machines or several threads.

Feature selection (FS) is a crucial preprocessing technique to deal with high-dimensional datasets that are common in the big data era. The feature selection technique's primary objective is to select a subset of features so that the selected subset contains discriminating power, the same as the original features set [1–3]. This technique can reduce the dimensionality of the feature space and improve classification performance due to removing irrelevant and redundant features [4].

According to the final result, feature selection techniques can be categorized into two subcategories: feature-subset selection (FSS) and feature ranking (FR). Moreover, depending on whether the label of each instance is available or not, the feature selection can be classified into supervised and unsupervised types [5–9].

Furthermore, according to dependency on a learning model, feature selection algorithms can be classified into three categories of Wrapper, Embedded, and Filter [10]. As it was mentioned by Brahim and Limam [11], the algorithms of the filter category have more generality than those of the other categories. Besides, these algorithms utilize a statistical criterion for feature evaluation resulting in a decrease of the computational cost.

Ensemble learning is based on the assumption that the combination of the output of several models obtains better results than the output of any individual models. Furthermore, ensemble learning algorithms have an inherent ability of distributing so that any base learners can be executed independently in separate workers or threads which is a useful characteristic in confronting a large scale dataset.

The ensemble learning broadly has been applied in the classification discipline in the last decade; however, its effectiveness is imaginable in other machine learning disciplines such as feature selection as well [12]. The ensemble learning approach for feature selection technique, which is called Ensemble Feature Selection (EFS), has received increased attention in recent years [13–17]. In another point of view, feature ranking as a subfamily of the feature selections is a common approach when the number of features is substantial. Hence, in this paper, the feature ranking algorithms (FRAs) are applied in EFS as basic learners and called ensemble feature ranking (EFR) henceforth. Furthermore, owing to what is known as the label of training instances, it is a supervised EFR.

To sum up, the EFR approach has three advantages: first, applying feature ranking has a lower computational cost than feature selection, and it is more sensible when the number of features is high. Second, the ensemble learning approach has inherent scalability because each base learning model can be processed independently, and this ability is essential to cope with large scale datasets. Third, we hope to make a more accurate result by applying several models instead of a single model.

Generally, an EFR has three steps. First, data diversity of training datasets is provided by a subsampling method, second, several rankings of features are determined by several base FR algorithms, and, third, the intermediate rankings are fused to generate the final ranking of the features. In the last decade, several rank fusion algorithms have been introduced in various scopes, such as opinion mining and information retrieval [18, 19].

The main purpose of this paper is studying the role and effectiveness of the different rank fusion methods as a part of homogeneous EFR approaches. For this purpose, seven FRAs belong to the filter category as base rankers combined with nine rank fusion methods in independent scenarios. Eventually, the experimental results are analyzed by statistical methods to respond to this question: is there a big difference in the rank fusion method applied in EFR? If so, which one can make more accurate results than others?

The remainder of the article is organized as follows: Section 2 introduces some background knowledge of the ensembles feature selection. Section 3 will describe the experimental framework, and, in Section 4, the experimental results will be presented. Section 5 offers some discussing remarks. Finally, concluding remarks will be given in Section 6.

2. Related Work

One of the most important techniques in data analyzing and processing is feature selection applied to broad scopes such as machine learning, pattern recognition, and data mining [20–24]. Furthermore, this technique can be more beneficial and sensible when the dataset is high-dimensional [25, 26]. The data often has lots of dimensions in some scopes, such as gene analyzing [27, 28], cancer classification [29], robotics [30], satellite images processing [31], and big data [32–34], which makes feature selection technique essential.

In the last years, several papers were published on EFS in different fields. In general, published articles can be classified into two groups. In the first group, the output of the proposed methods is a set of features, and, in the second group, it is a rank of features. In this section, ten state-of-the-art articles are considered so that the first three items belong to the first group and the rest belong to the second group.

A method called MCF-RFE was proposed by Yang and Mao [35], in which the outputs of several FR algorithms are fused to generate the final ranking of features. Then, the irrelevant features are removed by using the SVM-RFE algorithm. In this method, both the classification performance and the stability of the feature selection result are improved simultaneously. Although the proposed method used FR algorithms as base learners, its final output is a set of selected features.

In [36], Das et al. developed a method called EFSGA by applying a 7 biobjective genetic algorithm. Boundary region analysis of rough set theory and multivariate mutual information of information theory are employed as two objective functions in the proposed method. In their method, several subdatasets are prepared by the subsampling of the original dataset. Then, for each of the subdatasets, the biobjective genetic algorithm is executed, and several subsets of selected features are produced. Eventually, one of the selected feature subsets is determined as the final subset of features by using a heuristic method. It must be mentioned that both objective functions and also the genetic algorithm, as a population-based algorithm, are time-consuming and have high computational cost.

Hoque et al. in 2018 [37] proposed an algorithm called EFS-MI. EFS-MI applies some FR algorithms as base learners, and it tries to fuse the output of several FR algorithms at the final step. During the fusion step, the algorithm attempts to determine the final selected features set, so that it has both the maximum relation to class label and the minimum relation to the other features. This way causes the redundant features to have fewer chances to be a member of the final selected features set. The prominent note is that the proposed method uses the incremental approach at the fusing step that might reduce the distributing ability.

Despite both methods of MCF-RFE and EFS-MI, the base learners are of the FR type; their outputs were a set of selected features similar to the output of EFSGA. Some proposed methods, whose outputs are a feature ranking, are given in the following.

In the bioinformatics scope, an EFR method developed by Abeel et al. [38] uses the SVM-RFE as an incremental FR. In this paper, two rank fusion algorithms called CLA and CWA are introduced. In [39], a heuristic method is developed in which a given dataset is sent to 5 different FR algorithms. Then, based on the outputs of the FR algorithms, five classification models are made, and, eventually, the classifier's outcomes are combined by a simple voting. It is noteworthy that, in this method, the outputs of FR algorithms are not fused directly.

According to Brahim and Limam in 2013 [40], a fusion method named ROB-EFS for fusing the base feature rankings is introduced. In their method, the selected features are

fused based on two criteria of confidence and reliability. For assessing the confidence criteria, the method is applied to the SVM classification error rate that is thoroughly time-consuming.

Boucheham and Batouche proposed a method named MEFS for fusion feature ranking [41]. In this method, the feature rankings are fused in two steps. At the first step, the base feature rankings are generated in parallel, and then they are fused. In the second step, all actions of the first step are repeated incrementally.

A heterogeneous EFR algorithm is proposed by Seijo-Pardo et al. [42]. In this algorithm, all instances of a given dataset are considered by the six FR algorithms. Consequently, the six feature rankings are generated which are fused by the SVM-Rank for acquiring the final ranking of the features.

The authors of [40] published their next work [11] in which a new rank fusion method named RAA is proposed. The RAA method, similar to the Rob-EFS, utilizes the classification performance as a confidence criterion. Therefore, both articles have the same weaknesses.

3. Experimental Framework

In this section, the performances of different rank fusion algorithms applied in the homogeneous EFR algorithm are comparatively analyzed. Some aspects of experimental studies should be considered. These matters will be explained in the following.

3.1. Homogenous Ensemble Feature Ranking. In this paper, as mentioned before, rank fusion algorithms are applied in a homogeneous ensemble feature ranking, which is executed in a parallel approach. Therefore, explaining this algorithm is essential. In the EFR, at the first step, a given dataset is sampled S times and created S subdatasets. In the second step, each subsample is processed by an FRA in a separated thread independently, and then intermediate feature rankings are produced. At the final step, the intermediated rankings will be fused by utilizing a rank fusion algorithm to produce the final feature ranking. The details of the EFR are explained in Figure 1 and Algorithm 1. Note that, in Algorithm 1, $DS \in R^{N \times D}$ refers to the given dataset and the input variable S determines the number of local datasets that is assumed to 30 in the experimental studies. Also, the input variable I refers to the number of records in each local dataset, the input variable, A refers to an FRA that can be one of the algorithms introduced in Section 3.2, the input variable F refers to a rank fusion algorithm that can be one of the algorithms introduced in Section 3.3, and, finally, the output variable O refers to the feature ranking result.

3.2. Feature Ranking Algorithms as the Base Ranker. Seven FRAs are used in the EFR as base rankers. All these algorithms belong to the filter category, and they have a much lower computational cost than the algorithms that belonged to the wrapper and embedded categories. The

algorithms that belonged to the filter category are divided into multivariate and univariate methods. In general, one can say that most of the classical feature selection approaches are univariate; each feature is considered separately, having an important advantage in scalability, but at the cost of ignoring feature dependencies, thus perhaps leading to weaker performances than other feature selection techniques. In order to improve performance, multivariate filter techniques are proposed, but at the cost of reducing scalability [43]. The FRAs applied in experiments are listed. The five first of these algorithms are univariate, and two last are the multivariate algorithms. According to predefined statements, there is an expectation that the two last algorithms have a better performance than others, which is a salient point depicted in Figure 2.

- (i) Information gain: this criterion is based on the entropy measure that is used for feature ranking. The more the information gain value, the more important the feature [44–46].
- (ii) Gain ratio: this is a normal form of information gain. Although these ways are related to each other, the final feature rankings as outputs of these ways are different [47].
- (iii) Fisher: the main idea of Fisher score is to find a subset of features, such that, in the data space spanned by the selected features, the distances between data points in different classes are as large as possible, while the distances between data points in the same class are as small as possible [48–50].
- (iv) Gini: Gini-index is used to measure the impurity of a feature for categorizing. The smaller the value, i.e., the minor the impurity, the better the feature [51, 52].
- (v) OneRule: this method, named oneR, tries to build one simple rule to predict the target class for any feature, and then the algorithm sorts all the features based on the error rate of their rule. For example, a simple rule for a feature can be a set of feature values bound to their majority class [53].
- (vi) ReliefF: this method uses the ability of a feature in separating similar instances. For a random sample of the training set, the nearest hit and the nearest miss instances are found. Then, the algorithm updates the weight of all features based on the values of the hit and miss instances. Any feature that has a larger weight value can distinguish the instances of a class better [54–56].
- (vii) QPFS: in this method, features are evaluated by minimizing a multivariate quadratic function subjected to linear constraints. The quadratic function includes two components. The first one is a matrix of similarity among the features. The second one is a vector of dependency between the features and the target class. The result is a weight vector [57, 58].

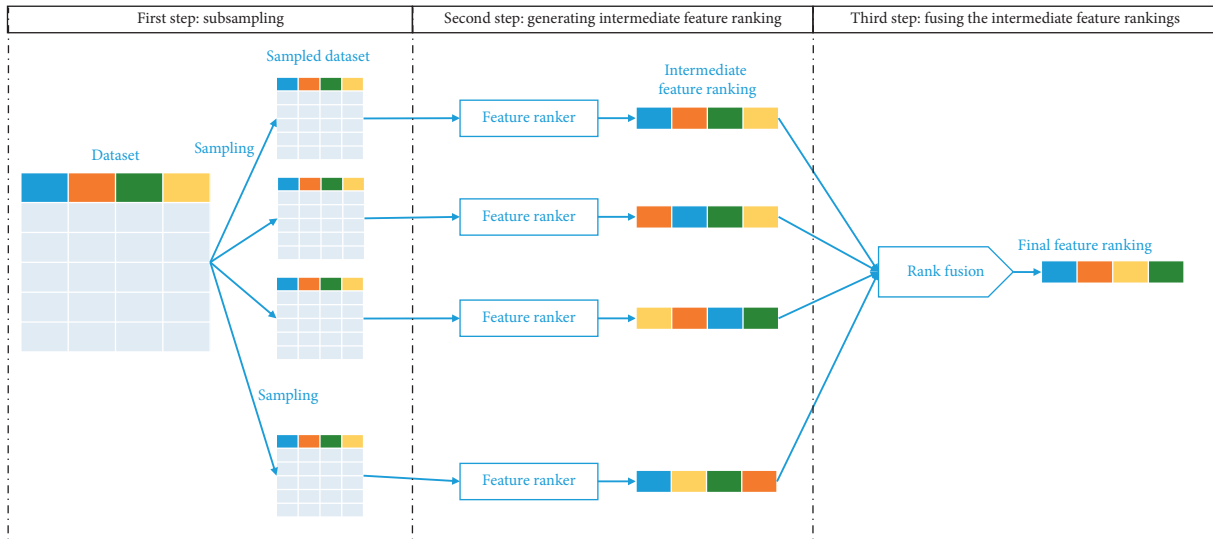


FIGURE 1: The scheme of ensemble feature ranking algorithm.

Input: $DS \in R^{N \times D}$ ▶ the given dataset
Input: S ▶ the number of sampling
Input: I ▶ the number of selected instances in a subsampled where $I \ll N$
Input: A ▶ the feature ranking algorithm
Input: F ▶ the rank fusion rank algorithm
Output: O ▶ the feature ranking result as the output

- (1) **for** $i = 0$ **to** S **do in parallel**
- (2) Create a local dataset $DS_{sub} \in R^{I \times D} \subset DS \in R^{N \times D}$ by subsampling
- (3) Send DS_{sub} to a worker node or worker thread
- (4) $IR_i =$ generate the intermediate feature ranking by applying A algorithm
- (5) **End for**
- (6) $O =$ fusion intermediate feature ranking $IR_{i=1, \dots, S}$ by applying F algorithm
- (7) **Return** O

ALGORITHM 1: Pseudocode of ensemble homogeneous feature ranking.

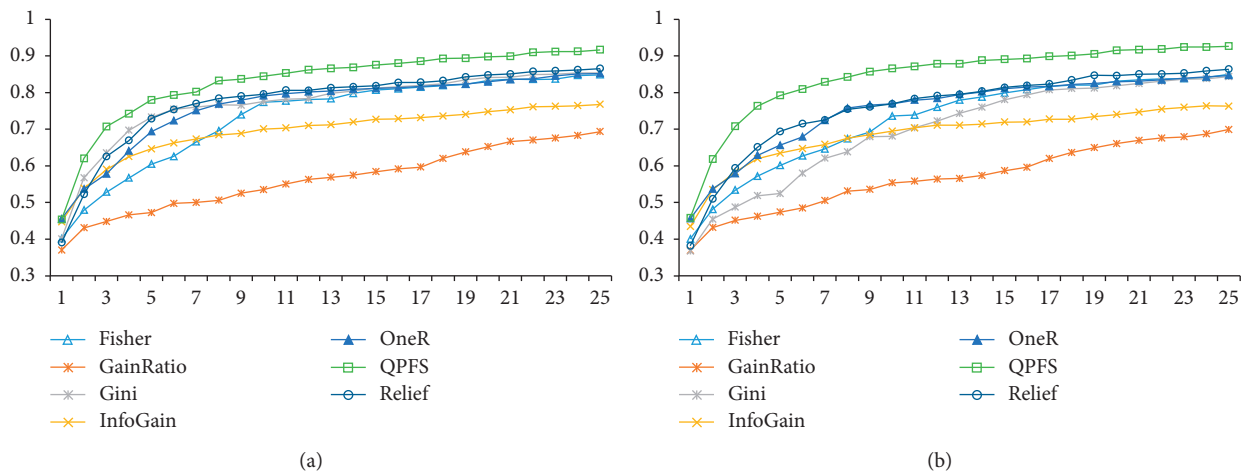
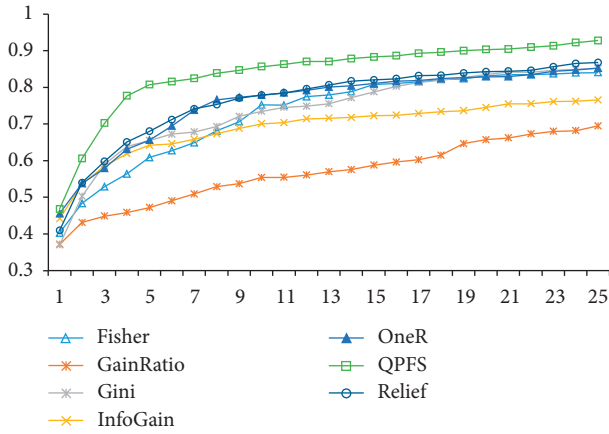
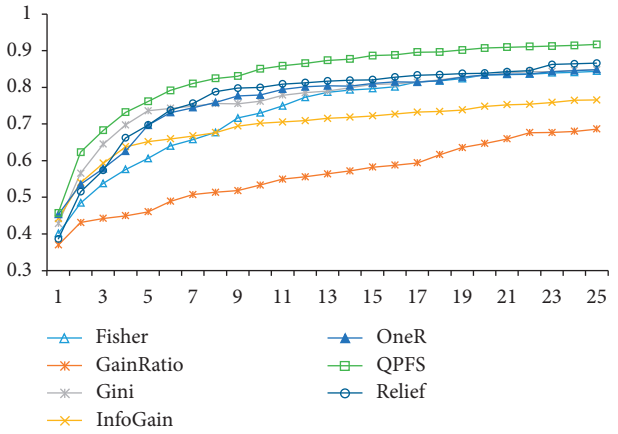


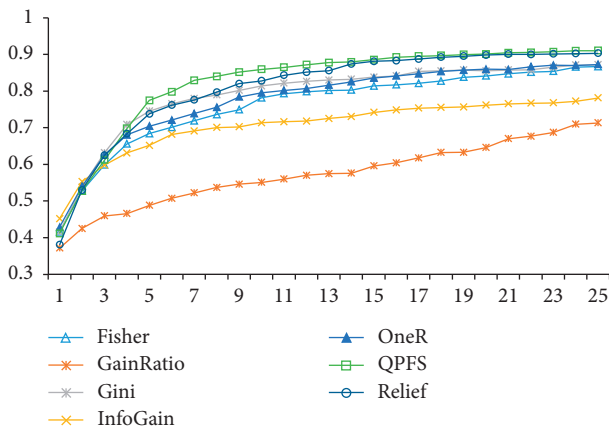
FIGURE 2: Continued.



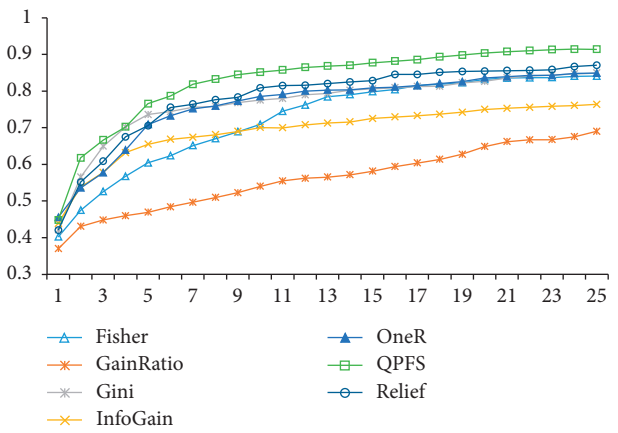
(c)



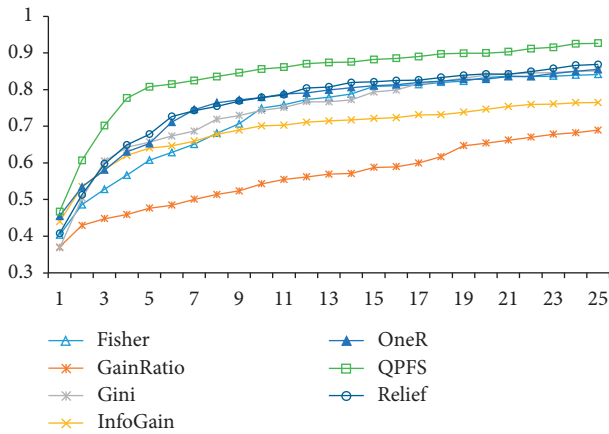
(d)



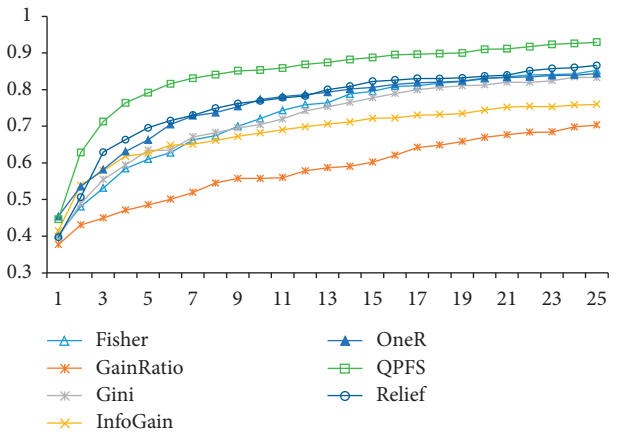
(e)



(f)



(g)



(h)

FIGURE 2: Continued.

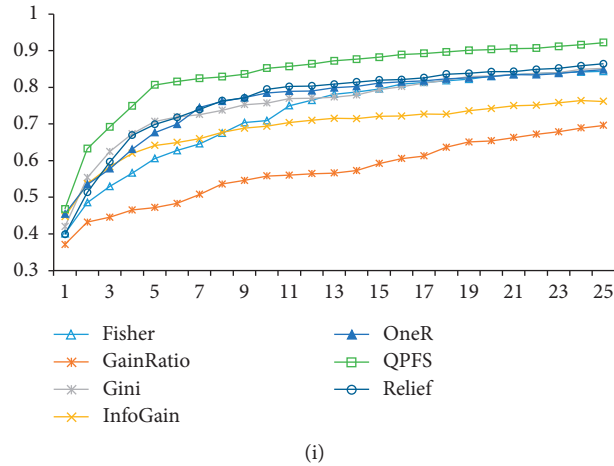


FIGURE 2: FFA curves, USPS dataset, and categorized based on rank fusion algorithms. (a) B-Geom. (b) B-L2. (c) B-Mean. (d) B-Median. (e) B-Min. (f) Kwik. (g) MC. (h) RRA. (i) Stuart.

3.3. Rank Fusion Algorithms. Rank fusion is called by different names such as rank aggregation, rank combination, and preference aggregation, in various scopes. Generating a final ranking by a set of base rankings is introduced as an optimization problem known as Kemeny ranking problem [59]. In the previous literatures, some algorithms, which are collected in Figure 3, were suggested for solving this problem [19, 60–63]. These solutions are categorized into two groups of exact and approximate solutions, although all of these algorithms are not suitable for applying in the EFR approach.

The exact solutions, such as integer linear program (ILP), have high computational cost and are time-consuming, so these methods are improper when the number of features in a given dataset is so large [62, 64]. For these reasons, none of the rank fusion algorithms that belong to the exact category are studied in this paper.

The approximate solutions have less computational cost than exact ones, so all rank fusion algorithms considered in the experiments belong to this category. Most of the rank fusion algorithms that belong to the positional category [65, 66] have low computational cost, are so fast, and are used in many different scopes. Therefore, in this paper, methods of Borda-Mean (B-Mean), Borda-Median (B-Median), Borda-Geometric-Mean (B-Geom), Borda-L2Norm (B-L2), and Borda-Min (B-Min) are studied. The rank fusion algorithms that belong to the sort based category [67–70] are not pretty high computational cost, despite being applied infrequently in the previous studies. Hence, just Kwik method as a representative of this category is studied.

The computational cost of the graph-based methods is not as high as the exact methods, although that is so noticeable still. In this paper, just the Markov-Chain (MC4) method is investigated because it has a better result than the other methods in the graph-based category [71]. Two rank fusion algorithms, robust rank aggregation (RRA) and Stuart [72, 73] that belong to the statistical order based, were applied a lot in the bioinformatics studies, and also the computational cost of these methods is lower than that of the

methods in the exact and graph-based categories. Thus, in experiments, both methods are studied.

Therefore, the computational cost, which is such a crucial factor in processing high-dimensional dataset, and popularity in the literature, are the two criteria to select fusion algorithms in experimental studies. Also, at least one fusion algorithm is selected except for the exact solutions category in each category. It is worth mentioning that the rank fusion algorithms, studied in the experiments, are distinguished by the bold font and blue colour as a mind-map in Figure 3.

3.4. Datasets. As mentioned before, the feature selection technique for dealing with a high-dimensional dataset is contemplated. For performing the experimental study, six datasets from two popular repositories, UCI (<https://archive.ics.uci.edu/ml/datasets.php>) and Scikit (<https://github.com/jundongl/scikit-feature/tree/master/skfeature/data>), are collected. These datasets have instances in a range of 1,440–21,048 and have features in a range of 179–1024, such that they are sampled 30 times in the execution time of the EFR algorithm. The characteristics of these datasets are gathered in Table 1.

3.5. Performance Assessment Criterion. In FR as a subcategory of FS, all features are ordered based on their importance and then by using a threshold; some more important features placed at the top of the ordered list are determined as the selected features. Notice that there is no deterministic way of determining the threshold value. Therefore, in this paper, to assess a feature ranking such as $R = (f_1, f_2, \dots, f_n)$, k top features are evaluated by using a stepwise method. The output of the stepwise evaluation method is a feature-forward-addition (FFA) curve [74]. The pseudocode for the stepwise evaluation method is given in Algorithm 2. Note that the input variable R is a ranking of features, the input variable k is the number of top features that is set to 10%, and the output variable O includes the FFA

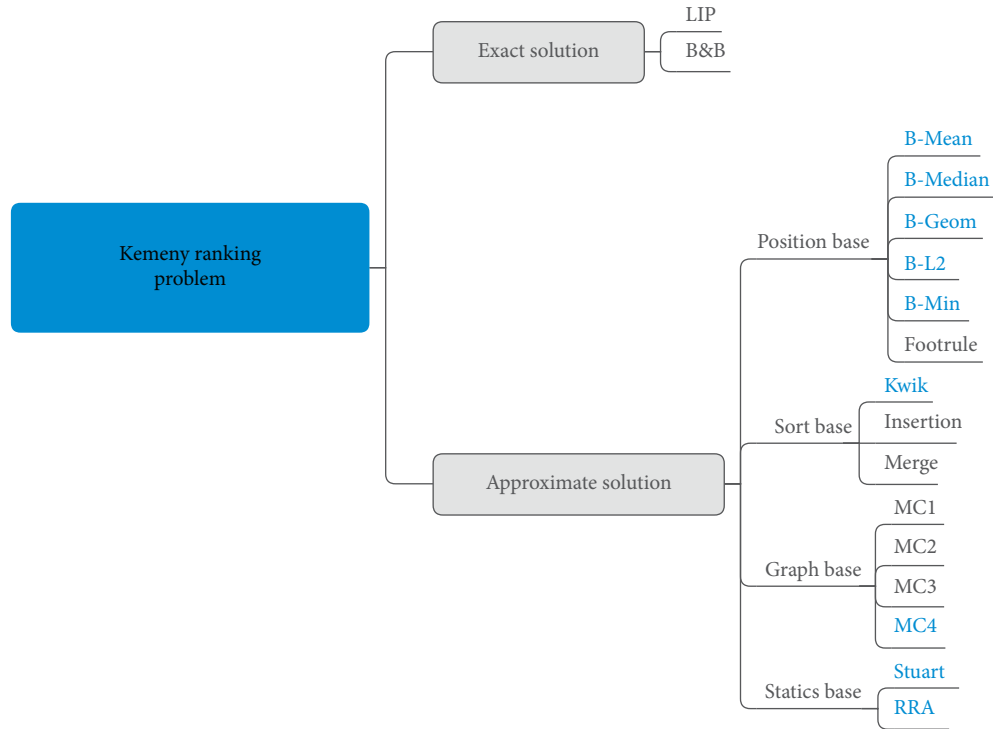


FIGURE 3: The categorization of rank fusion algorithms in various scopes.

TABLE 1: The characteristics of datasets used in experimental study.

#	Name	Instance#	Features#	Class#	Repository
1	Epileptic Seizure Recognition	11,500	179	2	UCI
2	USPS	9,298	256	10	Scikit
3	UJI (IndoorLoc)	21,048	521	3	UCI
4	HAR (Human Activity Recognition Using Smartphones)	10,299	561	6	UCI
5	Isolet	7,797	618	26	UCI
6	COIL20	1,440	1024	20	Scikit

curve points, and it is equal to the accuracy of a classifier when the various numbers of features (2 to 10% of all features) are selected. Also, the random forest classifier [75, 76] is applied for evaluation in this paper. The classifier’s implementation can be found in the Cran Repository (<https://cran.r-project.org/web/packages/randomForest/index.html>); its default settings, such as $n_{tree} = 500$, are used during experiments.

It is expected that the most important features which have more effect on the performance of the classification algorithm are placed at the top of a given feature ranking. Therefore, for two feature rankings, r_A and r_B , the better feature ranking reaches the maximum point on its FFA curve by a steeper slope, and, as a consequence, it has a larger area under the FFA curve, called AUFC henceforth. For example, in Figure 4, both feature rankings r_A and r_B have the same accuracy by using their fourteen top features, but it is obvious that r_A has better performance than r_B because it has a bigger area under the curve. Therefore, for assessing a feature ranking performance, the AUFC is used as a criterion [49]. A blue colour hatches the AUFC of r_B criterion in Figure 4.

4. Experimental Results

For performing the experimental study, some experiments are done on six real datasets whose characteristics were brought in Table 1. To this aim, nine rank fusion algorithms determined in Section 3.3, and seven FRAs introduced in Section 3.2 are applied in an EFR proposed in Algorithm 1 by 63 independent experiences (number of feature ranking methods \times number of rank fusion methods), for each dataset. Then, for each experience, the corresponding FFA curve is generated. Note that, for determining the training and test datasets, the fivefold cross-validation is used.

As an instance, Figure 2 depicts FFA curves for the USPS dataset, such that each subchart illustrates the FFA curve of various FRAs by applying a specific fusion method. As an example, the first subchart of Figure 2 illustrates FFA curves for FRAs and the B-Geom ranking fusion method when they are applied in an EFR.

As mentioned and expected, Figure 2 depicts that two multivariate algorithms, QPFS and ReliefF, have a better FFA curve than other methods in all subcharts. This matter is

```

Input:  $R$  ▶ a feature ranking such as  $f_1, f_2, \dots, f_n$ 
Input:  $k$  ▶ the number of top features
Output:  $O$  ▶ an array of FFA curve points such that  $|O| = k$ 
(1)  $S = \emptyset$  ▶ selected features
(2) for  $i = 1$  to  $k$  do
(3)    $S = S \cup R[i]$ 
(4)    $O[i]$  = accuracy of RandomForest classifier by using  $S$ 
(5) End for
(6) Return  $O$ 

```

ALGORITHM 2: Pseudocode of stepwise evaluation for the top- k feature ranking.

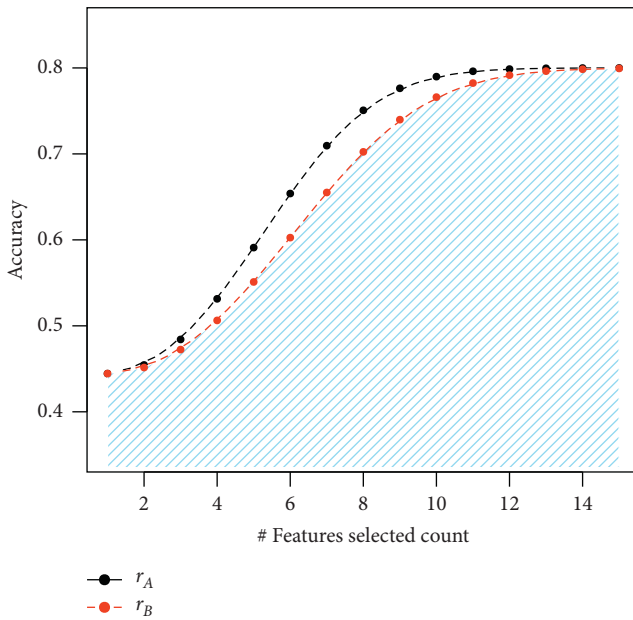


FIGURE 4: The FFA (Forward Feature Addition) curve and the AUFC (Area Under FFA Curve).

related to considering the relation between features in these algorithms, whereas other ranking algorithms evaluate features independently and disregard the dependency among them. It is evident that each dataset can generate a figure similar to Figure 2.

4.1. Statistical Comparison among Rank Fusion Algorithms. Nevertheless, the main purpose of this paper is to perform a comparative analysis of the performance of rank fusion methods applied in an EFR, so the FFA curves of Figure 2 should be categorized based on the FRAs, the same as what Figure 5 depicts. As illustrated by the Fisher chart in Figure 5, all rank fusion methods make almost similar FFA curves to each other when the Fisher algorithm is applied in an EFR; however, the B-Min method has a little better curve than others. By investigating the other subcharts, it can be seen that this behavior has been repeated among the other subcharts of Figure 5. Also, it is evident that each dataset can generate a figure similar to Figure 5. In order to perform the comparative analysis, the AUFC criterion is assessed for all FFA curves in Figure 5. Then, the AUFC measures are

normalized by dividing them on maximum AUFC (number of selected features \times the maximum point among all curve points). This process is repeated for all subcharts and for all six experiments datasets.

If the experimental results on each dataset are gathered, stacked column charts can be generated as Figure 6. As an instance, the second chart in Figure 6 is related to the performance of the B-Geom rank fusion method on the USPS dataset. This column includes the AUFC criterion values when the different FRAs are applied in the EFR in independent experiments, but all of these experiments applied the B-Geom method as the rank fusion algorithm. In other words, this column is a sum of normalized AUFC criterion values of the B-Geom curve in all subcharts of Figure 5. Therefore, the other columns in Figure 6 can be generated in the same way. The second chart in Figure 6 depicts that all of the rank fusion methods have the same performance; however, the B-Min method has a little better performance than the others. This behavior is repeated in the remaining subcharts in Figure 5.

For more deep analysis, the experimental results are collected in Table 2, such that the average value of each column of Figure 6 is placed in the equivalent cell in Table 2. As an instance, the average column values of the second chart in Figure 6 are placed in the second column in Table 2. The critical question is the following: is there a statistically significant difference in the average value of the AUFC among rank fusion algorithms when applied to an EFR?

To answer this question, at first, a one-way repeated measure analysis of variance (ANOVA) with Greenhouse-Geisser correction is conducted. The ANOVA test results reveal that there is a statistically significant difference in the mean AUFC values among rank fusion algorithms. At second, pairwise comparison t -tests, named PCT henceforth, by applying a Bonferroni adjustment, are performed to compare the mean AUFC values among rank fusion algorithms. The results of the pairwise comparison tests are shown in the columns in Table 2 by the letter-based representation method. For each column of the table, if there is no significant difference among rank fusion algorithms, the values are marked with a shared superscript letter. For example, in column USPS, the performance of the B-Min has a statistically significant difference with all of the other methods; consequently, it has none shared letter.

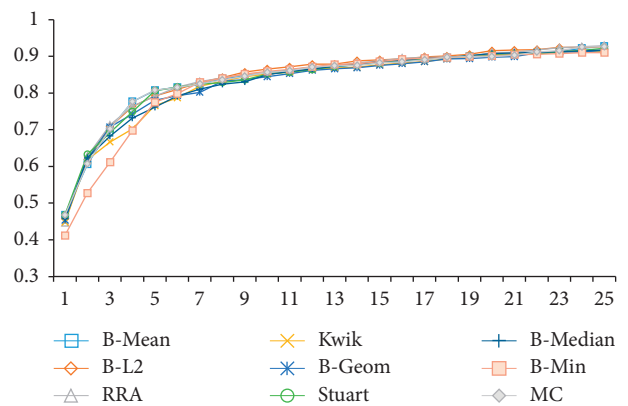
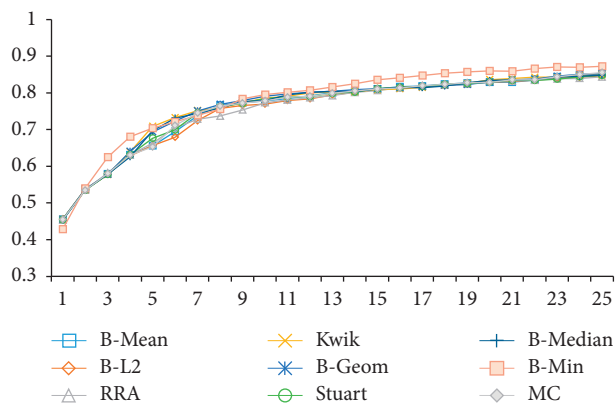
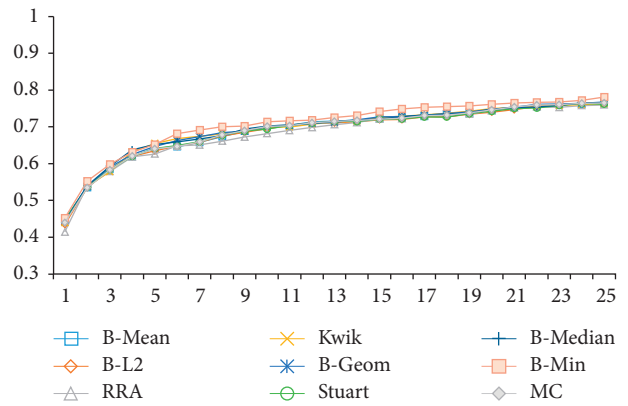
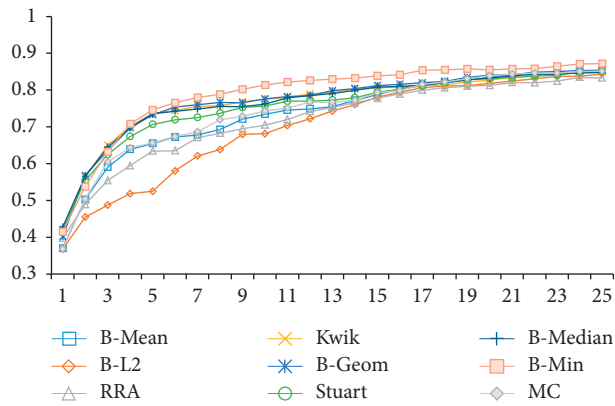
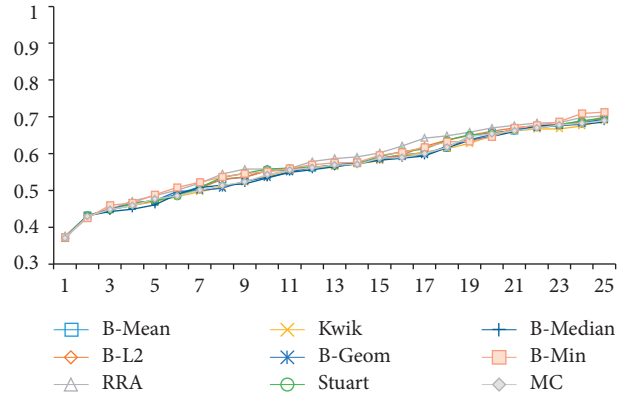
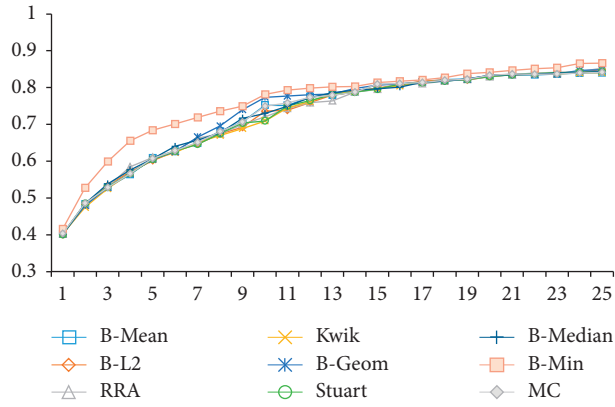


FIGURE 5: Continued.

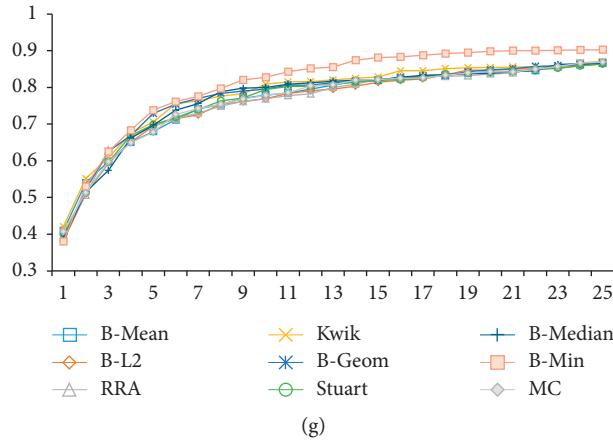


FIGURE 5: FFA curves for USPS dataset and different feature ranking and various fusion methods, categorized based on FR algorithms. (a) Fisher. (b) GainRatio. (c) Gini. (d) InformationGain. (e) OneRule. (f) QPFS. (g) ReliefF.

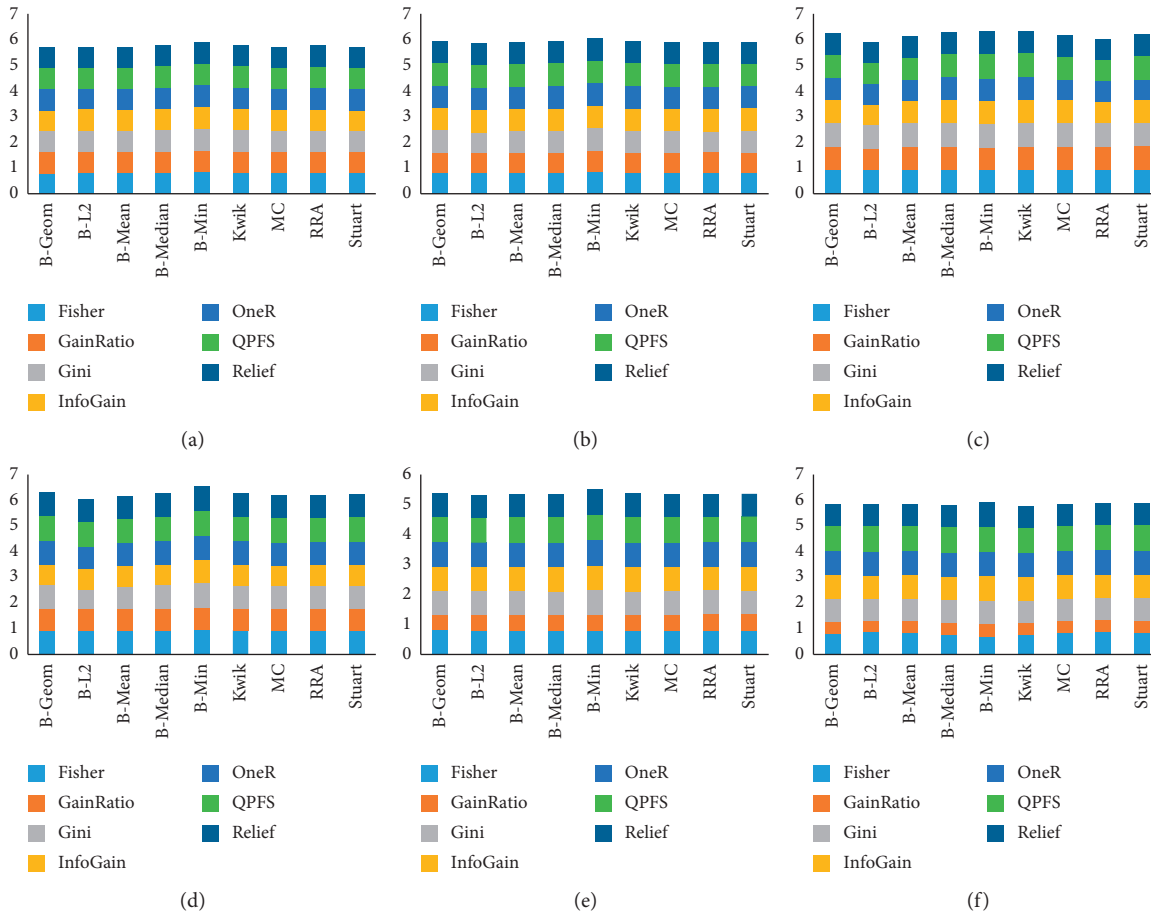


FIGURE 6: The stacked column charts of the AUFS criterion for each dataset. (a) Epileptic. (b) UPSP. (c) UJI. (d) HAR. (e) Isolet. (f) COIL20.

According to the first column in Table 2, the B-Min method has a better result than the others, and PCT realized that there is a statistically significant difference between the B-Min method and the others, so there is no shared letter. Consequently, in the Epileptic column in Table 3, the B-Min

method is set to 1 as the best fusion method in the Epileptic dataset. Also, the PCT does not realize the statistically significant difference among B-Median, Kwik, and RRA methods, and then all of these methods are set to 2 in the equivalent columns in Table 3.

TABLE 2: The performance average of the rank fusion algorithms based on AUFC criterion for all datasets.

Dataset	Fusion					
	Epileptic	USPS	UJI	HAR	Isolet	COIL20
B-Geom	0.820 ± 0.019 ^(a)	0.847 ± 0.039 ^(d)	0.909 ± 0.035^(c)	0.906 ± 0.067 ^(d)	0.759 ± 0.114 ^(c)	0.814 ± 0.176^(def)
B-L2	0.821 ± 0.018 ^(a)	0.833 ± 0.039 ^(a)	0.854 ± 0.053 ^(a)	0.867 ± 0.076 ^(a)	0.747 ± 0.109 ^(a)	0.810 ± 0.181^(abcdef)
B-Mean	0.821 ± 0.019 ^(a)	0.840 ± 0.037 ^(c)	0.888 ± 0.040 ^(f)	0.886 ± 0.067 ^(c)	0.752 ± 0.111 ^(b)	0.809 ± 0.180^(cd)
B-Median	0.830 ± 0.021 ^(b)	0.843 ± 0.040 ^(abc)	0.913 ± 0.031^(c)	0.905 ± 0.068 ^(d)	0.756 ± 0.114 ^(c)	0.802 ± 0.178^(abc)
B-Min	0.849 ± 0.015^(c)	0.866 ± 0.040^(e)	0.914 ± 0.025^(c)	0.941 ± 0.036^(f)	0.777 ± 0.117^(d)	0.830 ± 0.187^(abcdef)
Kwik	0.829 ± 0.018 ^(a)	0.844 ± 0.040 ^(abc)	0.915 ± 0.027^(c)	0.904 ± 0.067 ^(d)	0.754 ± 0.115 ^(b)	0.800 ± 0.181^(abc)
MC	0.820 ± 0.018 ^(a)	0.846 ± 0.037 ^(bd)	0.893 ± 0.038 ^(e)	0.890 ± 0.064 ^(b)	0.750 ± 0.111 ^(ab)	0.802 ± 0.184^(af)
RRA	0.827 ± 0.022 ^(ab)	0.840 ± 0.035 ^(bcd)	0.873 ± 0.056 ^(b)	0.893 ± 0.066 ^(bc)	0.751 ± 0.096 ^(bc)	0.818 ± 0.171^(be)
Stuart	0.822 ± 0.018 ^(a)	0.842 ± 0.036 ^(bc)	0.901 ± 0.039 ^(d)	0.900 ± 0.067 ^(e)	0.755 ± 0.105 ^(bc)	0.816 ± 0.171^(de)
STDEV	0.011	0.014	0.026	0.024	0.015	0.026

TABLE 3: The abstraction of Table 2, ranking of the rank fusion methods.

Fusion	Dataset						
	Epileptic	USPS	UJI	HAR	Isolet	COIL20	Average
B-Geom	3	2	1	2	2	1	1.83
B-L2	3	4	6	6	4	1	4.00
B-Mean	3	3	4	5	3	1	3.17
B-Median	2	3	1	2	2	1	1.83
B-Min	1	1	1	1	1	1	1.00
Kwik	2	3	1	2	3	1	2.00
MC	3	2	3	4	3	1	2.67
RRA	2	3	5	4	3	1	3.00
Stuart	3	3	2	3	2	1	2.33

In the USPS column of Table 2, the B-Min method has better performance than the other methods, and the PCT realized that this difference is statistically significant, so it is set to 1 in Table 3, and also both B-Geom and MC methods are set to 2 because there is no statistically difference between them.

In the third column in Table 2, the PCT could not recognize a significant difference between B-Geom, B-Median, Kwik, and B-Min; hence, all of these methods are set to 1 in the equivalent column in Table 3. Also, in HAR and Isolet columns in Table 2, there is a statistically significant difference between the B-Min method and the others, so this method is set to 1 in equivalent columns in Table 3. In contrast to the other columns, in the COIL20 column, the PCT could not realize a statistical difference between all methods. Therefore, all methods in the COIL20 column in Table 3 are set to 1. According to these explanations, Table 3 is filled, and it depicts that the B-Min method has a better result in all of the datasets, and it acquires the number 1 in the Average column.

In summary, Figure 6 illustrates that, in the EFR algorithm, there is a small difference in the performance of the rank fusion methods, and also Table 3 proves that the performance of the B-Min method has a statistically significant difference with most of the other methods.

4.2. EFR opposite to Individual FRA. In another point of view, investigating the performance of the EFR in a competition of the individual FRAs can be interested. To

this aim, Table 4 is generated that depicts the AUFC measures among EFR and individual feature rankings. As an example, the first cell in Table 4 shows the AUFC values of the EFR when the Fisher algorithm is applied as the base rankers on it, and the Fisher feature ranker is used individually. This cell depicts that the AUFC of the EFR (0.861) is bigger than the AUFC of individual Fisher (0.812). Note that highlighted values in bold style discriminate the better results in each cell.

As aforementioned in Section 1, the inherent scalability is the prominent advantage of the EFR to confront massive datasets due to independent processing of each subdataset by worker nodes or worker threads. Moreover, EFR can cause acquiring more accurate results than individual FRA owing to combining multiple models instead of a single model, and it is founded on the old proverb “two heads are better than one.” [12]. The latter is observable in Table 4. The results in this table illustrate that EFR has a proper potential to make more accurate results than individuals FRA ones, such that in 43/47 items the EFR has better results and in the remaining items 4/47 EFR has pretty comparable results with individual FRAs.

5. Discussion

Figures 5 and 6 and Table 2 illustrate that there is no big difference among rank fusion methods applied in a homogeneous EFR. This matter is related to two factors. First, the homogeneous ensemble approach applies a similar

TABLE 4: The comparison AUFC among ensemble feature rankings and individual feature rankings.

Approach	Base FRA											
	Epileptic		USPS		UJI		HAR		Isolet		COIL20	
	EFR	Individual	EFR	Individual	EFR	Individual	EFR	Individual	EFR	Individual	EFR	Individual
Fisher	0.861	0.812	0.877	0.835	0.940	0.937	0.961	0.958	0.818	0.797	0.872	0.837
GainRatio	0.849	0.816	0.813	0.766	0.934	0.922	0.885	0.815	0.574	0.540	0.492	0.456
Gini	0.861	0.812	0.905	0.874	0.940	0.937	0.964	0.923	0.815	0.760	0.948	0.704
InfoGain	0.868	0.823	0.903	0.854	0.915	0.916	0.918	0.785	0.840	0.804	0.960	0.938
OneR	0.865	0.817	0.892	0.878	0.899	0.923	0.957	0.946	0.863	0.811	0.972	0.955
QPFS	0.834	0.838	0.904	0.908	0.949	0.940	0.976	0.837	0.868	0.836	0.985	0.983
ReliefF	0.860	0.777	0.895	0.838	0.904	0.875	0.958	0.907	0.855	0.749	0.942	0.932

feature ranker algorithm as base rankers. Thus, all base feature ranking is generated based on a likewise procedure and logic. This matter can cause generating similar base feature rankings. Second, in a low noisy and real dataset, though, each sampled subdataset has different data instances, but there are no such diverse patterns among them. Therefore, applying similar feature ranker algorithms causes producing the base feature rankings whose most informative features placed on top positions are almost similar to each other, whereas the less informative features would be placed in various bottom positions in feature rankings. According to these two factors, various fusion algorithms make outcomes such that their top positions are almost similar.

In another point of view, Table 3 and statistical test results depict that B-Min method has a little better performance than others. The B-Min method uses an optimistic approach for the fusion of the base feature rankings. In an optimistic approach, a feature placed in the top positions at least in one base feature rankings would be a top member of the final feature ranking. In other words, in the optimistic approach, the base FRAs are assumed to be trustworthy, which is an efficient approach in noiseless and real datasets. Thus, most informative features of base feature rankings should have a decent chance to be a top placed informative feature in the final result. Generally, the experiments depict that the optimistic approach for the fusion of the base feature rankings which have almost similar top features can cause generating a bit better result than other fusion methods.

6. Conclusion

Feature selection is an essential preprocessing technique, and its importance is more sensible when the number of features of the given dataset is large that is an ordinary matter in the big data era. Also, the ensemble learning broadly has been applied in the classification discipline in the last decade; however, its effectiveness is imaginable in other machine learning disciplines such as feature ranking as well. The ensemble learning has inherent scalability due to the fact that each subdataset can be processed independently, and this ability is more important to cope with a large scale dataset. The EFR has three major steps, subsampling, generating intermediate feature ranking, and fusing the intermediate feature rankings. Because the fusing phase is a crucial step in the EFR, in this paper, a statistical analysis of

the performance of nine rank fusion methods is done when they are utilized in an EFR.

In the statistical analysis, a one-way repeated measure ANOVA with pairwise comparisons *t*-test applying a Bonferroni adjustment was performed to compare the mean of AUFC value among the rank fusion algorithms. The results of the one-way ANOVA revealed that the difference in the performance of the rank fusion methods is small, though there is a statistically significant difference in their performance when applied to the EFR algorithm. Additionally, the pairwise comparisons test showed that the “B-Min” method had a bit better performance than the other methods, at least on six real datasets that are examined in this paper.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

References

- [1] Y. Yang and J. O. Pedersen, “A comparative study on feature selection in text categorization,” in *Proceedings of the Fourteenth International Conference on Machine Learning*, Morgan Kaufmann Publishers Inc., Nashville, TN, USA, pp. 412–420, July 1997.
- [2] M. Soheili and A. M. E. Moghadam, “Feature selection in multi-label classification through MLQPFS,” in *Proceedings of the 2016 4th International Conference on Control, Instrumentation, and Automation (ICCIA)*, Qazvin, Iran, January 2016.
- [3] Y. Li, T. Li, and H. Liu, “Recent advances in feature selection and its applications,” *Knowledge and Information Systems*, vol. 53, no. 3, pp. 551–577, 2017.
- [4] I. Guyon and A. Elisseeff, “An introduction to variable and feature selection,” *Journal of Machine Learning Research*, vol. 3, pp. 1157–1182, 2003.
- [5] H. Zhang, R. Zhang, F. Nie, and X. Li, “An efficient framework for unsupervised feature selection,” *Neurocomputing*, vol. 366, pp. 194–207, 2019.
- [6] P. Zhu, W. Zuo, L. Zhang, Q. Hu, and S. C. K. Shiu, “Unsupervised feature selection by regularized self-representation,” *Pattern Recognition*, vol. 48, no. 2, pp. 438–446, 2015.

- [7] W. He, X. Zhu, D. Cheng, R. Hu, and S. Zhang, "Unsupervised feature selection for visual classification via feature-representation property," *Neurocomputing*, vol. 236, pp. 5–13, 2017.
- [8] P. Zhu, W. Zhu, Q. Hu, C. Zhang, and W. Zuo, "Subspace clustering guided unsupervised feature selection," *Pattern Recognition*, vol. 66, pp. 364–374, 2017.
- [9] R. Hu, X. Zhu, D. Cheng et al., "Graph self-representation method for unsupervised feature selection," *Neurocomputing*, vol. 220, pp. 130–137, 2017.
- [10] I. Guyon, S. Gunn, M. Nikravesh, and L.A. Zadeh, *Feature Extraction: Foundations and Applications (Studies in Fuzziness and Soft Computing)*, Springer, Berlin, Germany, 2006.
- [11] A. B. Brahim and M. Limam, "Ensemble feature selection for high dimensional data: a new method and a comparative study," *Advances in Data Analysis and Classification*, vol. 12, no. 4, pp. 937–952, 2017.
- [12] V. Bolón-Canedo and A. Alonso-Betanzos, "Ensembles for feature selection: a review and future trends," *Information Fusion*, vol. 52, pp. 1–12, 2019.
- [13] X. Xue, M. Yao, and Z. Wu, "A novel ensemble-based wrapper method for feature selection using extreme learning machine and genetic algorithm," *Knowledge and Information Systems*, vol. 57, no. 2, pp. 389–412, 2018.
- [14] B. Pes, "Ensemble feature selection for high-dimensional data: a stability analysis across multiple domains," *Neural Computing and Applications*, vol. 32, no. 10, pp. 5951–5973, 2019.
- [15] K. Benabdeslem, H. Elghazel, and M. Hindawi, "Ensemble constrained Laplacian score for efficient and robust semi-supervised feature selection," *Knowledge and Information Systems*, vol. 49, no. 3, pp. 1161–1185, 2016.
- [16] V. Kumar and S. Minz, "Multi-view ensemble learning: an optimal feature set partitioning for high-dimensional data classification," *Knowledge and Information Systems*, vol. 49, no. 1, pp. 1–59, 2016.
- [17] D. Tripathi, D. R. Edla, R. Cheruku, and V. Kuppili, "A novel hybrid credit scoring model based on ensemble feature selection and multilayer ensemble classification," *Computational Intelligence*, vol. 35, no. 2, pp. 371–394, 2019.
- [18] J. A. Balazs and J. D. Velásquez, "Opinion mining and information fusion: a survey," *Information Fusion*, vol. 27, pp. 95–110, 2016.
- [19] D. E. Losada, J. Parapar, and A. Barreiro, "A rank fusion approach based on score distributions for prioritizing relevance assessments in information retrieval evaluation," *Information Fusion*, vol. 39, pp. 56–71, 2018.
- [20] H. A. L. Thi, X. T. Vo, and T. P. Dinh, "Feature selection for linear SVMs under uncertain data: robust optimization based on difference of convex functions algorithms," *Neural Networks*, vol. 59, pp. 36–50, 2014.
- [21] K.-J. Wang, K.-H. Chen, and M.-A. Angelia, "An improved artificial immune recognition system with the opposite sign test for feature selection," *Knowledge-Based Systems*, vol. 71, pp. 126–145, 2014.
- [22] R. H. W. Pinheiro, G. D. C. Cavalcanti, and T. I. Ren, "Data-driven global-ranking local feature selection methods for text categorization," *Expert Systems with Applications*, vol. 42, no. 4, pp. 1941–1949, 2015.
- [23] S. Maldonado, R. Weber, and F. Famili, "Feature selection for high-dimensional class-imbalanced data sets using support vector machines," *Information Sciences*, vol. 286, pp. 228–246, 2014.
- [24] S. Maldonado, R. Montoya, and R. Weber, "Advanced conjoint analysis using feature selection via support vector machines," *European Journal of Operational Research*, vol. 241, no. 2, pp. 564–574, 2015.
- [25] D. Derroncourt, B. Hanczar, and J.-D. Zucker, "Analysis of feature selection stability on high dimension and small sample data," *Computational Statistics & Data Analysis*, vol. 71, pp. 681–693, 2014.
- [26] J. Yu, H. Zhong, and S. B. Kim, "An ensemble feature ranking algorithm for clustering analysis," *Journal of Classification*, vol. 37, no. 2, pp. 462–489, 2019.
- [27] Y. Saeys, I. Inza, and P. Larranaga, "A review of feature selection techniques in bioinformatics," *Bioinformatics*, vol. 23, no. 19, pp. 2507–2517, 2007.
- [28] C. Lazar, J. Taminau, S. Meganck et al., "A survey on filter techniques for feature selection in gene expression microarray analysis," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 9, no. 4, pp. 1106–1119, 2012.
- [29] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, "Gene selection for cancer classification using support vector machines," *Machine Learning*, vol. 46, no. 1/3, pp. 389–422, 2002.
- [30] T. T. Nguyen, Z. Li, T. Silander, and T.-Y. Leong, "Online feature selection for model-based reinforcement learning," in *Proceedings of the 30th International Conference on Machine Learning*, p. 506, Atlanta, GA, USA, June 2013.
- [31] P. Singh and G. Dhiman, "Uncertainty representation using fuzzy-entropy approach: special application in remotely sensed high-resolution satellite images (RSHRSIs)," *Applied Soft Computing*, vol. 72, pp. 121–139, 2018.
- [32] M. Soheili and A. M. Eftekhari-Moghadam, "DQPFS: distributed quadratic programming based feature selection for big data," *Journal of Parallel and Distributed Computing*, vol. 138, pp. 1–14, 2020.
- [33] R.-J. Palma-Mendoza, D. Rodriguez, and L. de-Marcos, "Distributed ReliefF-based feature selection in spark," *Knowledge and Information Systems*, vol. 57, no. 1, pp. 1–20, 2018.
- [34] R.-J. Palma-Mendoza, L. de-Marcos, D. Rodriguez, and A. Alonso-Betanzos, "Distributed correlation-based feature selection in spark," *Information Sciences*, vol. 496, pp. 287–299, 2019.
- [35] F. Yang and K. Z. Mao, "Robust feature selection for microarray data based on multicriterion fusion," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 8, no. 4, pp. 1080–1092, 2011.
- [36] A. K. Das, S. Das, and A. Ghosh, "Ensemble feature selection using bi-objective genetic algorithm," *Knowledge-Based Systems*, vol. 123, pp. 116–127, 2017.
- [37] N. Hoque, M. Singh, D. K. Bhattacharyya, and M. I. EFS-, "EFS-MI: an ensemble feature selection method for classification," *Complex & Intelligent Systems*, vol. 4, no. 2, pp. 105–118, 2018.
- [38] T. Abeel, T. Helleputte, Y. Van de Peer, P. Dupont, and Y. Saeys, "Robust biomarker identification for cancer diagnosis with ensemble feature selection methods," *Bioinformatics*, vol. 26, no. 3, pp. 392–398, 2010.
- [39] V. Bolón-Canedo, N. Sánchez-Marroño, and A. Alonso-Betanzos, "An ensemble of filters and classifiers for microarray data classification," *Pattern Recognition*, vol. 45, no. 1, pp. 531–539, 2012.
- [40] A. B. Brahim and M. Limam, "Robust ensemble feature selection for high dimensional data sets," in *Proceedings of the International Conference on High Performance Computing & Simulation (HPCS)*, Helsinki, Finland, July 2013.
- [41] A. Boucheham and M. Batouche, "Robust biomarker discovery for cancer diagnosis based on meta-ensemble feature

- selection,” in *Proceedings of the Science and Information Conference*, London, UK, August 2014.
- [42] B. Seijo-Pardo, V. Bolón-Canedo, I. Porto-Díaz, and A. Alonso-Betanzos, “Ensemble feature selection for rankings of features,” *Advances in Computational Intelligence*, Springer International Publishing, Cham, Switzerland, 2015.
- [43] V. Bolón-Canedo, N. Sánchez-Maróño, and A. Alonso-Betanzos, “Feature selection for high-dimensional data,” *Artificial Intelligence: Foundations, Theory, and Algorithms*, p. 147, Springer International Publishing, Cham, Switzerland, 2015.
- [44] C.-M. Lai, W.-C. Yeh, and C.-Y. Chang, “Gene selection using information gain and improved simplified swarm optimization,” *Neurocomputing*, vol. 218, pp. 331–338, 2016.
- [45] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, Wiley, Hoboken, NJ, USA, 2006.
- [46] H. Uğuz, “A two-stage feature selection method for text categorization by using information gain, principal component analysis and genetic algorithm,” *Knowledge-Based Systems*, vol. 24, no. 7, pp. 1024–1032, 2011.
- [47] H. Dag, K. E. Sayin, I. Yenidogan, S. Albayrak, and C. Acar, “Comparison of feature selection algorithms for medical data,” in *Proceedings of the International Symposium on Innovations in Intelligent Systems and Applications*, pp. 1–5, Trabzon, Turkey, July 2012.
- [48] G. Quanquan, L. Zhenhui, and H. Jiawei, “Generalized Fisher score for feature selection,” *CoRR*, <https://arxiv.org/abs/1202.3725>, 2012.
- [49] Q. Song, H. Jiang, and J. Liu, “Feature selection based on FDA and F-score for multi-class classification,” *Expert Systems with Applications*, vol. 81, pp. 22–27, 2017.
- [50] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, Wiley-Interscience, Hoboken, NJ, USA, 2nd edition, 2000.
- [51] A. Arauzo-Azofra, J. L. Aznarte, and J. M. Benítez, “Empirical study of feature selection methods based on individual feature evaluation for classification problems,” *Expert Systems with Applications*, vol. 38, no. 7, pp. 8170–8177, 2011.
- [52] L. Huan and Y. Lei, “Toward integrating feature selection algorithms for classification and clustering,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 4, pp. 491–502, 2005.
- [53] R. C. Holte, “Very simple classification rules perform well on most commonly used datasets,” *Machine Learning*, vol. 11, no. 1, pp. 63–90, 1993.
- [54] K. Kira and L. A. Rendell, “The feature selection problem: traditional methods and a new algorithm,” in *Proceedings of the National Conference on Artificial Intelligence*, p. 129, San Jose, CA, USA, July 1992.
- [55] H. Liu and H. Motoda, *Computational Methods of Feature Selection*, Chapman & Hall, Boca Raton, FL, USA, 2008.
- [56] M. Robnik-Šikonja and I. Kononenko, “Theoretical and empirical analysis of ReliefF and RReliefF,” *Machine Learning*, vol. 53, no. 1, pp. 23–69, 2003.
- [57] D. P. Bertsekas, *Nonlinear Programming*, Athena Scientific, Belmont, MA, USA, 1995.
- [58] I. Rodríguez-Lujan, R. Huerta, C. Elkan, and C. S. Cruz, “Quadratic programming feature selection,” *Journal of Machine Learning Research*, vol. 11, pp. 1491–1516, 2010.
- [59] J. G. Kemeny and J. L. Snell, *Mathematical Models in the Social Sciences*, MIT Press, Cambridge, MA, USA, 1978.
- [60] G. Nápoles, R. Falcon, Z. Dikopoulou, E. Papageorgiou, R. Bello, and K. Vanhoof, “Weighted aggregation of partial rankings using ant colony optimization,” *Neurocomputing*, vol. 250, pp. 109–120, 2017.
- [61] M. Khodabakhshi and K. Aryavash, “Aggregating preference rankings using an optimistic-pessimistic approach,” *Computers & Industrial Engineering*, vol. 85, pp. 13–16, 2015.
- [62] J. A. Aledo, J. A. Gámez, and D. Molina, “Tackling the rank aggregation problem with evolutionary algorithms,” *Applied Mathematics and Computation*, vol. 222, pp. 632–644, 2013.
- [63] B. Wang and J. Zhang, “The rank of fusion systems,” *Annali di Matematica Pura ed Applicata*, vol. 195, no. 1, pp. 29–39, 2016.
- [64] M. Meila, P. Kapil, A. Patterson, and J. A. Bilmes, “Consensus ranking under the exponential model,” in *Proceedings of the Twenty-Third Conference on Uncertainty in Artificial Intelligence*, AUAI Press, Vancouver, BC, Canada, July 2007.
- [65] J. Borda, *Memoire Sur Les Elections Au Scrutin*, Kessinger Publishing, Whitefish, MT, USA, Histoire de l’Academie Royal des Sciences, 1781.
- [66] D. Coppersmith, L. K. Fleischer, and A. Rurda, “Ordering by weighted number of wins gives a good ranking for weighted tournaments,” *ACM Transactions on Algorithms*, vol. 6, no. 3, pp. 1–13, 2010.
- [67] N. Ailon, M. Charikar, and A. Newman, “Aggregating inconsistent information,” *Journal of the ACM*, vol. 55, no. 5, pp. 1–27, 2008.
- [68] A. van Zuylen and D. P. Williamson, *Deterministic Algorithms for Rank Aggregation and Other Ranking and Clustering Problems*, Springer, Berlin, Germany, 2008.
- [69] A. Ali and M. Meilä, “Experiments with kemeny ranking: what works when?” *Mathematical Social Sciences*, vol. 64, no. 1, pp. 28–40, 2012.
- [70] F. Schalekamp and A. v. Zuylen, “Rank aggregation: together we’re strong,” in *Proceedings of the Eleventh Workshop on Algorithm Engineering and Experiments (ALENEX)*, pp. 38–51, New York, NY, USA, January 2009.
- [71] C. Dwork, R. Kumar, M. Naor, and D. Sivakumar, “Rank aggregation methods for the Web,” in *Proceedings of the 10th International Conference on World Wide Web*, pp. 613–622, ACM, Hong Kong, May 2001.
- [72] R. P. DeConde, S. Hawley, S. Falcon, N. Clegg, B. Knudsen, and R. Etzioni, “Combining results of microarray experiments: a rank aggregation approach,” *Statistical Applications in Genetics and Molecular Biology*, vol. 5, no. 1, 2006.
- [73] R. Kolde, S. Laur, P. Adler, and J. Vilo, “Robust rank aggregation for gene list integration and meta-analysis,” *Bioinformatics*, vol. 28, no. 4, pp. 573–580, 2012.
- [74] I. Slavkov, J. Karcheska, D. Kocev, and S. Dzeroski, “HMC-ReliefF: feature ranking for hierarchical multi-label classification,” *Computer Science and Information Systems*, vol. 15, no. 1, p. 43, 2018.
- [75] R. Genuer, J.-M. Poggi, C. Tuleau-Malot, and N. Villa-Vialaneix, “Random forests for big data,” *Big Data Research*, vol. 9, pp. 28–46, 2017.
- [76] L. Breiman, “Random forests,” *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.