A New Feature Filtering Approach by Integrating IG and T-Test Evaluation Metrics for Text Classification

Abubakar Ado[1], Mustafa Mat Deris[2]
Noor Azah Samsudin[3]
Faculty of Computer Science and Information Technology
Universiti Tun Hussein Onn Malaysia
Batu Pahat, Malaysia

Aliyu Ahmed[4]
Department of Computer Science
Bauchi State University
Gadau, Nigeria

***Abstract*—High dimensionality is one of the main issues associated with text classification, such as selecting the most discrepant features subset for classifier's effective utilization is a difficult task. This significant preprocessing stage of selecting the relevant features is often called feature selection or feature filtering. Eliminating the non-relevant and noise features from the original feature set will drastically reduce the size of the feature set and the time complexity of the classification models and also improve or maintain their performance. Most of the existing filtering method produced a subset with relatively high number of features without much significant impact on running time, or produced subset with lesser number of features but results in performance degradation. In this paper, we proposed a new bi-strategy filtering approach that integrates Information Gain with t-test that selects a subset of informative features by considering both the score and ranking of respective features. Our approach considers the results' disparity produced by the benchmark metrics used in order to maximized and lessen their advantage and disadvantage. The approach set a new threshold parameter by computing V-score of the features with minimum scores present in both the two subsets and further refined the selected features. Hence, it reduces the size of the features subset without losing much informative features. Experiment results conducted on three different text datasets have shown that the proposed method is able to select features that are highly discrepant and at the same time achieves a significant improvement in terms of classification accuracy and F-score at the cost of a minimum running time.**

Keywords*—Dimensional reduction; feature filtering; feature selection; t-test; information gain; V-score

I. INTRODUCTION

In this emerging era of computing and internet technology, especially the emerging of social media, text analytic becomes more cumbersome[1]. As a result, both the size of features and instances of a textual dataset has been increasing rapidly. The increasing size of the text data results in diverse research problems to text analytic tools, such as machine learning. Text classification is one of the pronounce problem associated with text analytics [2][3], and currently is becoming one of the most vital research direction in the field of machine learning.

Text classification or documents classification is the problem of assigning unlabeled text instances to one or more predefined labelled classes or categories [4] [5][6][7]. Text classification has been utilized in various application domains [8], e.g. spam filtering [9], Sentiment Analysis [10], Natural Language Processing [11][12] Information Retrieval, Text Mining and so on.

One of the most crucial steps of the preprocessing of text data is the presentation of text documents into vector space via Bag_of_Word (BOW) [13][14][15]. The final product of this task is associated with two main issues, a vast number of features representation, and the presence of irrelevant and noisy features which general termed high dimensionality[15]. These issues can cause a lot of problems for the Text classification task, which is known to be intrinsically high dimensional [4] [5]. Classification in a situation that involves high number of features or high-dimensional space can become infeasible or very difficult due to computational complexity expensiveness [16][17]. However, feature reduction approach is considered as a dimensional reduction problem. The huge features generated introduces the so-called *"dimensionality curse*" with thousands of features that increase the computational complexity of a classifier [18][19][20][21][22]. Curse of dimensionality is a popular known problem for machine learning models [23]. When it arises in text classification, it seriously worsens the performance of the classifier in terms of classification accuracy and running time [5][24].

The main goal of dimensionality reduction is to reduce the number of features without worsening the performance of the classifier [14] [19]. As the key way to overcome this problem, feature selection (FS) technique can be applied to filter out irrelevant, redundant and noisy features and selects the most informative subset of features from the original features set [1][19]. This task will aggressively reduce the original vectors space representation of features into lower-dimensional vector representation [25][26][27]. Moreover, the properties of the informative features in the original feature set would be unaltered in the processes of dimensionality reduction. Feature selection (FS) approach ranks the original features according to some criterion evaluation (scores) and selects the top-ranked features to form an informative subset [27], which retains a good degree of discriminating capability in separating documents of various categories [28][29]. In contrast to the feature selection, feature extraction approach transforms the text documents on to a new lower-dimensional space from their original high dimensional feature instead of selecting a features subset from the original features set [30][31] [15].

Generally, feature selection methods [32] are broadly grouped into filter methods, wrapper methods[33], and embedded methods [34][35][27]. Filter methods [36] are independent that they do not interact with classifier when constructing an informative features subset. They rely on metrics for evaluating and ranking the importance of a feature prior to the classification. The methods can attain quick feature sorting to effectively filter out a high number of non-relevant or noise features [27]. They select features subset by considering the usefulness of a feature according to evaluation metrics [35][28][27][37]. Filter methods usually have good computational efficiency but sacrifice classification accuracy to some extent. Information Gain [38], Chi-Square [39], Fisher Score [40], ReliefF [41], *t*-test [4] are among the few filter based methods. Wrapper methods are dependent on classifiers that they frequently interact with the classification algorithm in order to construct a subset of informative features [13][35][27]. They evaluate a particular feature subset by training and testing a given classifier. The methods are tailored to a particular classifier [42]. These methods have bad computational efficiency but result in high classification accuracy, and they are not usually favoured in text classification task [43]. Heuristic Search Algorithms (HSA) and Sequential Selection Algorithms (SSA) [44][45][46] are common examples of classical wrapper methods. Embedded Methods integrate classifiers with feature selection technique during the training phase and optimally search feature subset by designing an optimization function [35][44][47]. Like wrapper methods, embedded methods frequently interact with the classifier but have computational efficiency better than wrapper methods, and are also tailored to a specific classifier [43]. Selection-Perceptron (FS-P)[48], Support Vector Machines (SVM-RFE) [49], Lasso (L1) and Elastic Net (L1+L2) based models [50][51] are some few examples of embedded based methods.

This paper is based on filter FS approach, and goal of this research work is to propose a new approach that selects more informative features from the original features set which help classification model to achieve good performance with regard to both time complexity and classification accuracy. The main point of view is on dimensional reduction, to reduce the number of features and processing time without sacrificing the classification accuracy. The features are exposed to double filter-based evaluation metrics (IG and t-*test*), in which at the final output, are obtained, only the discriminate features that highly contribute to the classification task, and produce a lower dimensionality subset base on features' respective rank and score. The approach blends the concepts of intersection and vector magnitude to select a subset of refined informative features by considering both the score and ranking of respective features. An experiment conducted with three distinct text datasets has shown that the proposed approach produces acceptable results by achieving a recorded performance of 67.65%, 54.74%, and 80.16%, and running time of 7464ms, 4689ms, and 29806ms on 20NewsGroups, NewsCategory, and Reuters-21784, respectively. This shows that the method retains most of the informative features when compared with other chosen methods.

The remaining body of this paper is systematically partitioned as follows: In Section 2, related works are presented. The proposed approach and the Filter-based feature selection methods employed explicitly by the approach, namely IG and t-test are discussed in Section 3. Properties of the datasets used and experimental set up are devoted to Section 4. Experiment results and discussion are systematically placed in Section 5. Finally, the study ends with a conclusion and highlights of possible future work which are given in Section 6.

II. Related Works

There are large number of research works on filter-based feature selection metrics to remove irrelevant and noisy features in text classification problem. The primary aim is often to reduce the feature dimensionality so as to minimize the processing time without sacrificing or improving the classification accuracy. In an effort to reduce the computational complexity, some numerous current works hybridized multiple scoring metrics to select most informative features. Results discrepancy is among the top challenges in hybridization approach as different results would be obtained when applying different evaluation metrics on the same dataset [38], and this issue can result in selecting noncontributory features. In this section, we will briefly present some review of those works, and lastly, we will summarize the drawbacks of the existing methods.

Lewis [52] uses mutual information (MI) to measure the importance of a feature, thus proposed a new scoring metric known as Mutual Information Maximization (MIM) that computes the relevancy between *n* features and classes. Liu and Setiono [39] proposed an algorithm that computes the score of each feature and selects relevant features based on chi-square score. The algorithm calculates the numeric attribute intervals and selects features according to the statistical data characteristics. A comparative study by Mladenic and Grobelnik [53] on a different dataset was conducted, and only for the Multinomial Naïve Bayes (NB) model upheld Odds Ratio over a wide variety of evaluation metrics been compared. For feature filtering, Bi-Normal Separation (BNS) has previously been described to be outstanding in ranking terms. Forman [54] improve an existing scoring metric for features by substituting IDF with BNS. The new method, TF-BNS scales the magnitude values and rank features by computing the BNS score of every feature. Empirical evaluation of text classification tasks using Support Vector Machine (SVM) shown significantly better performance in terms of F-measure and accuracy. Uguz [55] applies IG to ranked terms in a given document according to their importance in the initial stage of his proposed framework. Vinh et al. [56] proposed a new approach for selecting feature by normalizing well known MI (Mutual Information) measurement and used it to assess the potentiality of the features. Despite the competitive results achieved, the proposed approach could not conceal the highly correlated features influence the classification outcomes. Azhagusundari and Thanamani [57] developed a feature selection method based on IG for selecting the discriminant features from a give original set. The authors used IG to build a discernibility matrix which could be used to select the optimal subset of

features from the set of original data. Experimentally they showed their method obtained comparative classification accuracy on comparison with the original dimensionality. A greedy feature selection method using mutual information is introduced by Hoque and et al. [58]. The method blends feature–feature and feature–class MI to select the optimal feature subset. Wang et al. [4] use the concept of term frequency and developed a new feature scoring metric approach based on t-test, the method measures the diversity of the distributions of a feature between the particular category and the entire dataset. Experiment results indicate that the proposed method is marginally better than IG and chi-square method in terms of micro-F1 and macro-F1. Rehman et al. [5] proposed a novel function metric for feature ranking named Normalized Differences Measure (NDM), which evaluate the rank of a term by considering the term's relative document frequencies in both positive and negative classes. Zhou et al. [37] proposed a feature selection algorithm that uses segmented term frequency to compute the frequency of a document. Moreover, the impact of the same feature term to the classification under the dissimilar frequency of term is deeply considered. The algorithm uses the resultant terms' frequencies to give scores to each available feature and selects those features that are above a defined threshold. When Compared with six different FS methods, the empirical result demonstrated that the proposed method could able to increase classification accuracy on a textual dataset.

All the works mentioned earlier are single FS methods that consider only a single strategy for the selection of an informative subset of features. Consideration of multiple strategies altogether is impossible with a single feature selection method. In view of that, the hybridization approach has received significant attention in the field of dimensional reduction currently. The methods combined different FS methods considering various aspects of the features into single. Tsai and Hsiao [59] combine multiple methods for dimensional reduction to figure out more informative features for stock prices prediction task. The method integrates decision tree, PCA, and genetic algorithm as search methods, and utilizes the concept of an intersection, union, and multi-intersection approaches to filter out irrelevant variables. An intermediary method of union (OR) and Intersection (AND) approach named modified union is presented by Bharti and Singh [60]. The authors applied union (OR) and intersection (AND) on *k*-top selected ranked features, and on remaining unselected features subset, this merges the feature subsets into a single subset and further select the most relevant features. The feature filtering methods used in the study are document frequency (DF) together with term variance (TV). To exploit the advantages of two different FS methods, a hybridization of cluster-based and the frequency-based approach is presented by Nguyen and Bao [13]. The proposed method termed FCFS on comparison with its counterpart achieved the best performance in terms of micro-F1. To tackle the problem of results discrepancies, a new feature selection approach that combines the computed scores from multiple FS methods into one is proposed by Rajab [61]. The proposed method normalizes and computes vector score (V-Score) magnitude of each feature using the scores produced based on IG and Chi-square function metrics, and selects the top-ranked features. Kamalov and Thabtah [62] proposed a method that selects optimal features from sets with ranking features produced by three different ranking strategies. The authors used vector scores (V-Scores) to stabilize the scores obtained from three methods (IG, Chi-square, and inter-correlation) and assign a new rank to each feature. To further remove non-relevant and noise features from feature subsets produced by two different evaluation functions, Li et al. [27] consider the application of union approach on the lowest rank feature subset produce by Fisher score and IG methods.

Many studies have investigated the strength of several filtering methods and their combination in the literature. Forman [32] empirically studied and compared twelve different evaluation metrics for feature selection on a text classification problem, and they finally revealed that BNS with IG has the minimum correlated failure so as mark best backup choice. The impact of integrating five methods for FS was investigated by Thubaity et al. [63]. The study employed IG, Chi-square, NGL, GSS, and RS methods on Arabic textual dataset. Union (OR) and intersection (AND) approach were utilized to integrate the scores produced from various FS methods employed to a single sorted feature set. Results Analysis showed there was no any improvement recorded in terms of classification accuracy when more than three FS metrics were integrated, while a small improvement was noticed for integrating two to three FS metrics. Vora and Yang [64] present a comparative study on ten different filtering methods namely Fisher Score, Chi-square, Gini Index, Laplacian Score, IG, mRmR, CFS, FCBF, Kruskal-Wallis, and REliefF. Experimented on five different text dataset, the authors found that combination of Kruskal-Wallis, Gini Index with SVM classifier lead the race as it achieved competitive classification performance but takes longer processing time, while IG and Chi-2 are projected as methods with a large number of similar features have been selected.

III. Material and Method

This section presents a brief discuss on the information gain and t-test algorithm since both are useful for the proposed approach that will be explained in sub-section *C*.

A. Information Gain Algorithm

Gain (IG) [38][65], is an information theoretical and entropy-based method which is widely used in the field of dimensional reduction [43] [37]. IG is previously used to determine attribute use in splitting instances in decision tree-based models [66] and currently is applied to select the informative features subset in a given set of features. The method computes and assigns score to each feature considering the variation between entropy obtained based on presence or absence of term in a given category [37]. High information gain or high score indicates the discriminating capability of a feature and ranked top. The entropy of discrete random variable X is formulated as:

$$H(X) = -\sum_{x_i \in X} P(x_i) \log\big(P(x_i)\big) \quad (1)$$

x_i denotes a specific event of the variable$X$, $P(x_i)$ denotes the probability of an event (x_i). The general formula for computing IG of a given feature *t* is given as:

$$IG(t,c) = \sum_{c\in\{c_i,\bar{c}_i\}} \sum_{t\in\{t_k,\bar{t}_k\}} P(t,c) * \log\left(\frac{P(t,c)}{P(t)*P(c)}\right) \quad (2)$$

where *P(t, c)* is the probability of class *c* and occurrence of the feature *t*. *P(t)* is the probability of class containing feature *t*, *P(c)* is the probability of class *c*. $\overline{t_k}$ *and* $\bar{c}_k$ denote feature not present, and class not present, respectively. Let *N* represent the total number of documents in a given dataset, and *Ns* with indicated subscripts values represents counts of documents. Using Maximum Likelihood estimates (MLEs) of probabilities, equation (2) can be expressed as:

$$\mathrm{I(t,c)} = \frac{\mathrm{N_{11}}}{\mathrm{N}}\log_2\left(\frac{\mathrm{NN_{11}}}{\mathrm{N_{1.}N_{.1}}}\right) + \frac{\mathrm{N_{01}}}{\mathrm{N}}\log_2\left(\frac{\mathrm{NN_{01}}}{\mathrm{N_{0.}N_{.1}}}\right)$$
$$+ \frac{\mathrm{N_{10}}}{\mathrm{N}}\log_2\left(\frac{\mathrm{NN_{10}}}{\mathrm{N_{1.}N_{.0}}}\right) + \frac{\mathrm{N_{00}}}{\mathrm{N}}\log_2\left(\frac{\mathrm{NN_{00}}}{\mathrm{N_{0.}N_{.0}}}\right) \quad (3)$$

In information theory logic, a term/feature contains about the class, if the distribution of a term is equivalent in the class as it is in the whole collection, then $I(\mathrm{t,c}) = 0$. IG attains its optimal value if the term is a perfect discriminator for class membership if the term exists in a document if only the document is in the class.

B. Student Statistical Test Algorithm

Statistical Test (*t*-test) is a statistical-based method which is commonly used to evaluate if the means of two groups are statistically different from each other by computing a ratio between the mean difference of two groups and the variability of the two groups [4][67]. Presently, *t*-test is widely used as an evaluation function to select significant features that contribute to classifying instances. The method computes score of feature by measuring the distinct distributions of the term in relevant category and documents collection [68]. The formula for calculating *t*-test is given as:

$$t-test(t_i,c_k) = \frac{|\overline{tf_{k\iota}} - \overline{tf_\iota}|}{m_k \times s_i} \quad (4)$$

$$S_i^2 = \frac{1}{N-K}\sum_{k=1}^{k}\sum_{j\in C_k}(tf_{ij} - \overline{tf_{k\iota}})^2 \quad (5)$$

$$m_k = \sqrt{\frac{1}{N_k} - \frac{1}{N}} \quad (6)$$

Each class's specific scores obtained from (4) are combined to find the final score as follows:

$$t-test_{avg}(t_i) = \sum_{k-1}^{k} t-test(t_i, C_k) \quad (7)$$

where S_i denotes the standard deviation within a category, C_k denotes the k^{th} category, N_k is the number of documents in k^{th} category, k is the total number of categories, tf_{ki} denotes the average TF of term t_i in category k, tf_i donates average TF of term t_i in the corpus. *N* is the total number of documents.

However, when the score is less than the defined threshold, it indicates that the feature has lower discrimination ability; otherwise, the feature will contribute in the classifying instances and will be selected.

C. Proposed Approach

Considering the problem of result discrepancy produced when two filtering methods are combined, and the risk of losing informative features, an approach is proposed named new bi-strategy feature filtering approach which hybridizes IG with t-test to remove indiscriminate features by taking into consideration both feature ranking and vector score magnitude (*V-score*). The approach applies IG and t-test metrics independently to compute scores and assign the computed scores to each feature in the original features set, let say D_1 and D_2. The top-ranked features that are greater than a predefined threshold K_1 are considered as significant features and are selected, new subsets of features S_1 and S_2, which are based on IG and *t*-test are generated independently. Next, a feature with minimum IG score from S_1 and a feature with minimum *t*-test score from S_2 that are present in both S_1 and S_2 are selected and their V-scores are computed. The minimum V-score among the two computed V-scores is set as the new threshold K_2. The approach further refines the features subsets by selecting a feature only if it is present in S_1 or S_2 and its V-score is greater than the new defined threshold K_2 otherwise it is an indiscriminate feature and will be neglected.

V-score of a given feature is computed using the concept of vector magnitude proposed in [61] that is, summing the squares of a vector's coordinates and taking the square root of the summation, it is formulated as:

$$V_{score} = \sqrt{(IG_{score})^2 + (t-test_{score})^2} \quad (8)$$

NB: The values of the scores produced by IG is different from that of *t*-test. So, we have to normalize the scores first before computing V-score so as to uniformly transform them into equivalent scale.

Let us consider Table I below, which contains few samples extracted from 20NewsGroups. It shows generated ranking of each feature based on the chosen filter methods. It can be seen that there are presence of discrepancies in the output. This issue arises due to the different theoretical strategy used by distinct filter methods to compute the score of each feature in the given dataset. IG ranked "Thanks" the lowest while *t*-test ranked it the highest, there is high assurance for IG method to eliminate this particular feature when a threshold is defined despite it has been selected by *t*-test method as the most informative feature. Therefore, both methods fall into the problem of losing informative features, likewise the existing hybrid filtering methods. Nevertheless, the proposed approach mitigates such issue by considering both the ranking and score of each feature. The approach sets a new thresholds base on computed V-score and further refines the features subset.

TABLE I. RANKING PRODUCED BY IG AND T-TEST FILTER METHODS

Features	**Ranking**	
	IG	***t-test***
space	1	3
god	2	2
orbit	3	5
religion	4	6
people	5	4
thanks	6	1

Framework of the proposed approach is based on the following algorithm 1:

ALGORITHM 1: PROPOSED BI-STRATEGY FILTERING APPROACH "MIN_MAX_V-SCORE"

INPUT:
- D: Set of documents with N features
- c: Set of label classes
- K_1: Initial threshold

OUTPUT:
- SL: Subset of selected features

FEATURE_SCORING(D, c)
1. $L_1 \longleftarrow [\,]$, $L_2 \longleftarrow [\,]$
2. T $\longleftarrow$ EXTRACT TREMS IN DOCUMENTS (D)
3. **For each** t_i in T **do**:
4. $A(t_i, c) \longleftarrow$ **COMPUTE**(SCORE(t_i, c)) using eqe(1) through eqe(3)
5. APPEND($L_1(A(t_i, c), t_i)$)
6. $A(t_i, c) \longleftarrow$ **COMPUTE**(SCORE(t_i, c)) using eqe(4) through eqe(7)
7. APPEND($L_2(A(t_i, c), t_i)$)
8. **End For**
9. **SORT**(L_1), **SORT** (L_2)
10. **Return** L_1, L_2

Begin
1. $L_1, L_2 \longleftarrow$ **FEATURE_SCORING**(D, c)
2. FS1 $\longleftarrow k_1\%\ \{L_1\} = \{t_1, \ldots t_q\}$
3. FS1 $\longleftarrow k_1\%\ \{L_2\} = \{t_1, \ldots t_n\}$
4. j $\longleftarrow$ q
5. **For** t_i **in** [**SORT**.**descend**(FS1)]
6. Normalise (t_i)
7. **If** $t_i \in \{FS1\} \cap \{FS2\}$
8. $V_{score}(1) \longleftarrow$ **COMPUTE**(V_{score}_of_t_i using eqe (8))
9. **Break**
10. **For** t_j **in** [**SORT.descend**(FS2)]
11. Normalise (t_j)
12. **If** $t_i \in \{FS1\} \cap \{FS2\}$
13. $V_{score}(2) \longleftarrow$ **COMPUTE**(V_{score}_of_t_j using eqe(8))
14. **Break**
15. $k_2 \longleftarrow$ MIN($V_{score}(1)$, $V_{score}(2)$)
16. **APPEND** [SL, {FS1∩FS2}]
17. **For** t_i **in** [{$L_1 \cup L_2$} – {SL}] **do**
18. **If** $V_{score}(t_i) >= k_2$
19. **APPEND**[SL, (t_i)]
20. **End For**
21. **Return** SL

End

A summarized flowchart of the proposed methodology is depicted in Fig. 1. The process begins with the raw datasets as input. After relatively balancing the all unbalanced datasets, then original features set is constructed using TF-IDF. Next step is the initial features subsets formation using IG and t-test filter methods to compute and assign a score to all features and a sequence of high ranked features will be selected. Next, we employed the proposed BI-strategy filtering approach to further refined the initial features subsets and generate the new informative features subset. Lastly, we validate the new approach by recording the Classification accuracy and f-score of the selected classifiers.

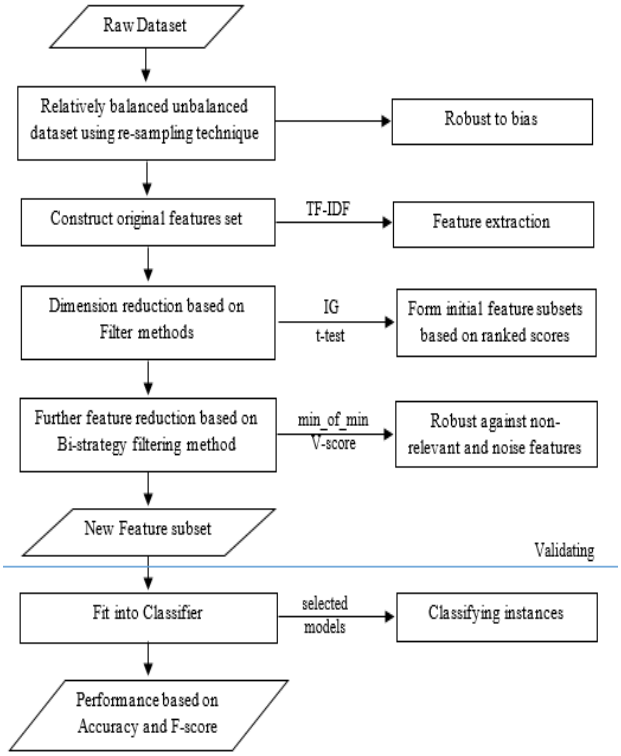

Fig. 1. Flowchart of the Proposed Methodology.

D. Experiment and Datasets

In this sub-section, summary of the datasets used and the experimental process adapted are briefly explained. The classification algorithms employed are also presented. Lastly, the section ends with a discussion on classifiers and implementation requirements.

1) Dataset: To evaluate the proposed approach in this experiment, the well-known three text benchmark datasets widely used for multi-class classification task is selected. Two of the datasets (Reuters 21578 and News Category) are unbalanced while the other one (20newsGroups) is balanced. We believed that both the datasets are highly dimensional with large number of samples, and also diversity amount of classes is considered. The summary information of the datasets is display in Table II.

TABLE II. SUMMARY OF THE DATASETS USED

Dataset	#Instances	#Features	#Classes
20news Groups	18846	173451	20
News Category	140597	1268350	36
Reuters-21578	11367	16578	90

The 20NewsGroups approximately comprises of 20,000 documents gathered from the collection of Usenet Newsgroups [69], and it consists of relatively balanced 20 distinct categories, each category contains around 1000 documents. The Reuters-21578 comprises of 21578 documents gathered from Reuters newswire, and it consists of unbalanced 135 categories with each document is associated with at least one categories (multi-label) [70][60]. Before importing the dataset into our experiment, we assigned only one category label to each document by stripping out all country names on the list and selecting the first topic left. Moreover, any document that is not associated with any topic was also eliminated from the dataset. This significantly reduced the number of categories and documents to 90 and 11367. The News Category comprises of around 150, 000 samples gathered from Short News Category, and it consists of unbalanced 41 categories with each document is associated with at least one categories (multi-label). We combined some few categories that can be naturally merged together, such as 'CULTURE & ARTS', ARTS & CULTURE', and 'ART'. We finally reduced the number of categories to 36.

2) Experiment settings: In the initial phase of the experiment, all English letters are converted into lowercase, stop words are removed, and words having non-characters are filtered. After then, roots of English words are found by applying porter stemmer algorithm [71]. And lastly, feature extraction is performed using TF-IDF weighting [72]. NB: all the three datasets are randomly divided into 60% training and 40% testing.

To validate the proposed filtering approach and its effectiveness on classification models, two existing benchmark methods for feature filtering are selected for comparison, namely IG and t-test. The selection is based on the fact that the proposed method is a hybrid of the selected methods. Besides the new approach, three other existing hybrid filtering approaches include Union (OR) approach, Intersection (AND) approach and Vector Magnitude (V-score) approach proposed in [61] are also selected. The initial threshold value K_1 is based on the number of features been ranked in the original set and was set as 60% for both IG and t-test, any feature below the predefined threshold is low scored feature and will be disregarded otherwise will be qualified for further selection evaluation. Jaccard Similarity Coefficients (JCC) is used in this study to measure the similarity of features been selected by different benchmark filtering methods.

Five different well-known classification methods are used for validation purpose in this study. The selection is based on the positive recommendation of the methods in terms of text multi-class classification. The selected methods including Support Vector Machine (SVM) [18][22], Naïve Bayes (NB) [73], Decision Tree (DT) [38], Random Forest (RF)[74][18], and Ridge Regression (RR) [75][76]. All these models will be used to record the classification accuracy and performance of the stated filtering methods. Default values of most of the parameters associated with the classification methods are retained. For SVM and NB, multi-class SVC with kernel function and MultinomialNB are adapted while for RF number of estimation was set to 100 when executed on 20News groups and Reuters 21578 datasets and set to 20 on News Category dataset, respectively.

Because of space limit, the performance of the classification methods will be reported using two standard recognized metrics widely used for text classification in literature, namely, Accuracy and F1-score. Accuracy is the percentage of the documents that are classified correctly in the given entire documents dataset. F1-score is the representation of harmonic mean of precision and recall. Accuracy and F1-score ware computed using the following equations.

$$accuracy = \frac{TP+TN}{TP+FP+TN+FN} \quad (9)$$

$$f-score = \frac{2\times P\times R}{P+R} \quad (10)$$

Where, $'TP'$ = True Positive (documents correctly classified as positive), $'TN'$ = True Negative (documents correctly classified as negative), $'FP'$ = False Positive (documents incorrectly classified as Positive), $'FN'$ = False Negative (documents incorrectly classified as Negative), and $'P'$ and $'R'$ are precision and recall values and are computed using the following equations.

$$precision = \frac{TP}{TP+FP} \quad (11)$$

$$recall = \frac{TP}{TP+FN} \quad (12)$$

In this study, all the implementations for the experiment are conducted on Python (V3.8.2) environment, which is installed on a computer with Windows 8 (OS). Other minimum required conditions for the experiment include Intel(R) Core™ i5 processor4300m@2.60GHz/8GRAM/64 GB.

IV. Discussion of Results

Table III shows a brief description of the chosen filtering methods based on the formulation and strategy adapted. As it can be seen that all the selected hybrid filtering method ware formulated by integrating information theory and statistical theoretical based benchmark methods (IG and *t*-test), the reason behind the selection of this two benchmark methods is by considering the Jaccard Similarity Coefficients between them which is very low compared to other methods. The average percentages of features reduced by different methods in all the three datasets are displayed in Fig. 2. From the figure, it can be seen that the percentage reduction differences in terms of feature dimensions between the proposed method (PM) and existing methods (IG, TS, UA, IA, VS). PM, UA, and IA reduced the number of features by 52.07%, 19.2% and 60% on 20NewsGroups Dataset, where as 48.08%, 26.53%, and 53.48% on NewsCategory Dataset and finally 45.25%, 28.86% and 51.15% on Reuters-21578 Dataset respectively. While IG, TS and VS reduced the features by 40% in all the three datasets, this is because a fixed threshold K_1 was defined in all the experiments. The figure reveals in all the three datasets, the proposed method comparatively reduces the feature dimensions, with IA and UA achieved the highest and lowest percentage of features been reduced in all the datasets.

TABLE III. A BRIEF DESCRIPTION OF THE EXISTING AND PROPOSED FILTERING METHOD(S) EMPLOYED FOR THE EXPERIMENT

Method	Acronym	Description	Strategy
Information Gain	IG	Selects top-ranked k features based on IG scores: {K%(IG)}	Single
t-test	TS	Selects top-ranked features based on t-test score: {K%(TS)}	Single
Union Approach	UA	Selects features from hybrid of IG and TS based on Union (OR) approach: {K%(IG)} ∪ {K%(TS)}	Hybrid
Intersection Approach	IA	Selects features from hybrid of IG and TS based on intersection (AND) approach : {K%(IG)}∩ {K%(TS)}	Hybrid
V-Score	VS	Selects features from hybrid of IG and TS based on V-score: {K%(VS)}	Hybrid
Proposed Approach	PM	Selects features from hybrid of IG and TS based on modified V-score: {K_2%VS({K_1(IG) ∪ K_1(TS)})}	Hybrid

The proposed approach could not beat IA method in terms of feature reduction because we seriously take into consideration the risk of avoiding losing informative features which will suffer the performance of classifier as discovered with IA and related methods. In particular, our proposed approach saves as an intermediary between IA and the other methods.

Tables IV, V and VI show the classifiers' performance including Ridge, MNB, SVC, DT and FR based on classification accuracy and running time after applying the existing and proposed filtering methods on the text datasets. Best results are face bolded. The impact of filtering methods on the classifier performance in both the five classifiers results is noticeable.

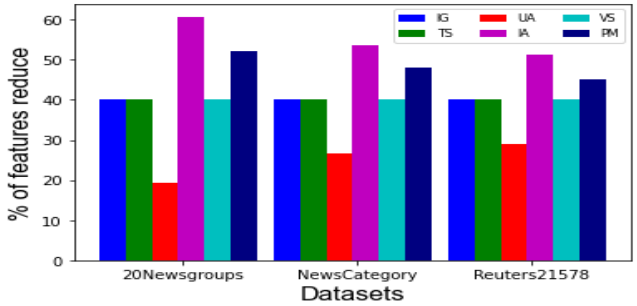

Fig. 2. Percentage of Features Reduced for 20 New Groups, News Category, and Reuters-21578 Datasets.

TABLE IV. PERFORMANCE OF THE EXISTING AND PROPOSED FILTERING METHOD(S) IN TERMS OF ACCURACY AND RUNNING TIME ON NEWS CATEGORY DATASET

Classifier	Metrics	PM	IG	TS	UA	IA	VS
Ridge	Accuracy	**0.59657**	0.59413	0.59604	0.59553	0.54383	0.59455
	Running Time (ms)	7340	7568	7818	8239	**7234**	7756
Multinomial NB	Accuracy	**0.54583**	0.53742	0.54318	0.53990	0.52575	0.53918
	Running Time (ms)	1223	1249	1287	1302	**1215**	1242
LinearSVC	Accuracy	**0.60230**	0.60001	0.60226	0.59994	0.54972	0.59913
	Running Time (ms)	14259	15527	15621	15855	**14117**	15497
Decision Tree	Accuracy	**0.45740**	0.45415	0.45214	0.45655	0.42503	0.45668
	Running Time (ms)	91011	96414	96382	102909	**90715**	97462
Random forest	Accuracy	**0.53541**	0.52881	0.52657	0.53331	0.51271	0.53361
	Running Time (ms)	120613	125019	124444	131862	**120578**	126953

TABLE V. PERFORMANCE OF THE EXISTING AND PROPOSED FILTERING METHOD(S) IN TERMS OF ACCURACY AND RUNNING TIME ON 20NEWSGROUPS DATASET

Classifier	Metrics	PM	IG	TS	UA	IA	VS
Ridge	Accuracy	**0.75362**	0.75203	0.75216	0.75266	0.73381	0.74973
	Running Time (ms)	0859	1064	1054	1173	**0791**	1071
Multinomial NB	Accuracy	0.74770	0.74000	**0.75008**	0.74177	0.72638	0.73134
	Running Time (ms)	0.035	0047	0055	0095	**0034**	0046
LinearSVC	Accuracy	0.74923	0.74832	**0.7538**	0.74460	0.72621	0.74159
	Running Time (ms)	0757	0977	1001	1063	**0742**	0931
Decision Tree	Accuracy	**0.47285**	0.47930	0.47054	0.47064	0.44798	0.47170
	Running Time (ms)	8032	9552	10765	8268	**7958**	8797
Random forest	Accuracy	0.65909	0.65759	**0.66289**	0.65015	0.64113	0.65051
	Running Time (ms)	27641	29985	30004	31452	**27602**	30043

TABLE VI. PERFORMANCE OF THE EXISTING AND PROPOSED FILTERING METHOD(S) IN TERMS OF ACCURACY AND RUNNING TIME ON NEWS CATEGORY DATASET FOR DIFFERENT CHOSEN FILTERING METHODS ON 20 NEWSGROUPS DATASET

Classifier	Metrics	PM	IG	TS	UA	IA	VS
Ridge	Accuracy	**0.84774**	0.84697	0.84667	0.84579	0.83403	0.84579
	Running Time (ms)	2788	3114	3158	3160	**2615**	3023
Multinomial NB	Accuracy	**0.80123**	0.79420	0.79683	0.80299	0.74947	0.80064
	Running Time (ms)	0074	0.089	0092	0084	**0074**	0084
LinearSVC	Accuracy	**0.84902**	0.84667	0.84755	0.84667	0.80872	0.84667
	Running Time (ms)	1180	1324	1348	1356	**1144**	1311
Decision Tree	Accuracy	**0.72619**	0.72032	0.72618	0.72527	0.67178	0.72589
	Running Time (ms)	2822	2935	2974	3045	**2820**	2995
Random forest	Accuracy	**0.78511**	0.78100	0.77602	0.77631	0.72157	0.78130
	Running Time (ms)	7676	7901	7880	7942	**7676**	7899

In Table IV, accuracy results and running times are summarized for different chosen filtering methods on 20 Newsgroups dataset. Both the methods showed good performance with all the classifiers except with DT. The average classification accuracy, when filtering methods (IG, TS, UA, IA, VS and PM) were applied are 67.54%, 68.09%, 67.20%, 65.51%, 66.90%, and 67.65%. However, from the average score, we notice that the accuracy by different filtering methods is basically at the same level across all the classifiers with TS achieved the highest accuracy score followed by our method with no much significant difference. As can be seen from the table, in terms of running time, the proposed approach and AI marked the lowest as they achieved an average running time of 7464ms and 7425ms, thus supersede TS and the other methods. In particular, our method shows competitive performance on 20NewsGroups dataset.

The classification performance on NewsCategory dataset is shown in Table V. the average classification accuracy for the filter methods are 54.29%, 54.40%, 54.50%, 51.14%, 54.46%, and 54.75%. We notice the average accuracy of the proposed approach is comparatively little bit higher than that of the other filter methods compared. From the table, it can be seen that in most cases, our approach achieved a lower running time (4689ms averagely) but a little bit higher than IA (4635ms averagely). However, the overall comparison on NewsCategory dataset shows our method achieved significant performance.

Table VI reports the classification performance on Reuters-21578 dataset. It shows that the accuracy by different filtering methods is roughly similar. The average accuracy for the filter methods are 79.78%, 79.87%, 75.72%, 80.00%, and 80.16% . Compared with the other filter methods, the average accuracy of the proposed method is comparatively higher. The lowest average running time is achieved by IA as 29865ms and then followed by our method as 29806ms upon all the filter methods.

In general, the performance of the filter methods reported on each dataset is roughly at the same level across all the five classifiers. On 20NewsGroups dataset, we observed that TS recorded the highest classification accuracy with a slice difference than that of our method, but the running of our method is significantly lower than that of TS. While on NewsCategory and Reuters-21576 datasets, our method recorded the highest classification accuracy with lower running time. IA generally recorded the lowest running time upon all the classifiers but sacrificed their performance. This indicates that the method filters out some informative features, thus reducing classification capability. On the other hand, IG, TS, UA, and VS achieved competitive performance but the running time is significantly high, and this indicates the presence of noise and irrelevant features in the final subset produced which need to filter out so as to reduce the time complexity. We also observed that the best accuracy results were obtained with the Ridge classifier and SVC classifier, whereas the results that are obtained with DT are comparatively bad. Generally, the proposed method achieves acceptable performance on all datasets, which indicates that this kind of filter method can not only reduce the size of the features set but also ensure that informative features are retained so that the performance of a classifier is not sacrificing.

Despite work done to balance the two unbalanced datasets used in this experiment still, the datasets are relatively unbalanced. Therefore using accuracy metrics to evaluate the performance could be misleading. In order to further verified the validity of the proposed approach, Fig. 3, 4 and 5 shows the performance results based on F-score of the chosen classifiers when the proposed and existing filter methods were applied on the three datasets selected. Examining both the figures, we can see that the results obtained are in line with accuracy results obtained in Tables IV, V, and VI. The information depicted in Fig. 3 shows the approach recorded the highest F-score after TS with a relatively small difference. Moreover, in Fig. 4 and 5, the proposed approach attains the highest F- score with a minimal gap. Therefore, we conclude that the proposed method achieves the best classification performance in terms of the highest F-score on most of the cases. Although the proposed approach, does not always give the highest result on all the datasets such as with 20NewsGroups, but the F-scores results are still acceptable.

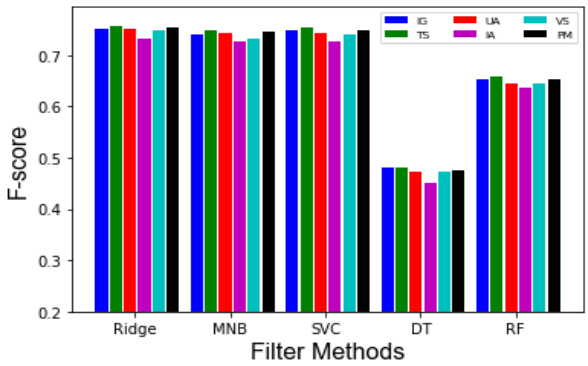

Fig. 3. Performance of the Existing and Proposed Filtering Method(s) in Terms of F-Score on 20 News Groups Dataset.

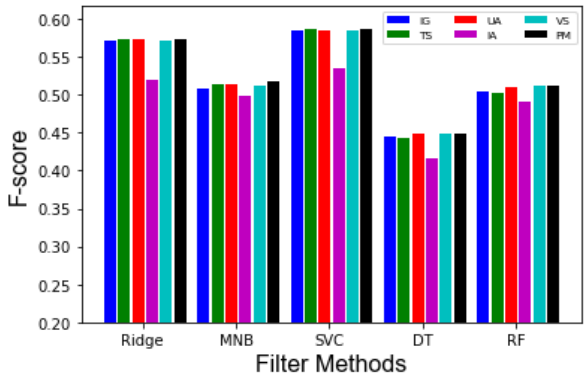

Fig. 4. Performance of the Existing and Proposed Filtering Method(s) in Terms of F-Score on News Category Dataset.

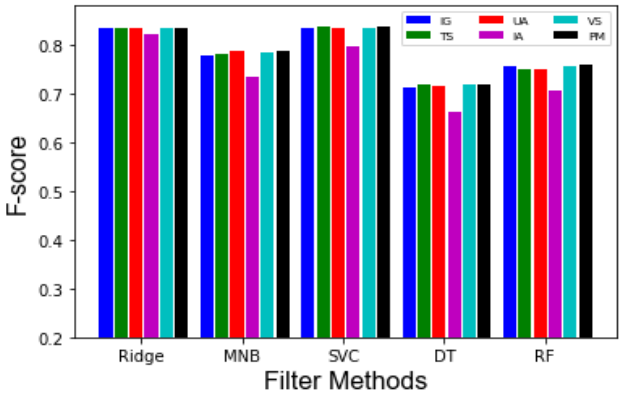

Fig. 5. Performance of the Existing and Proposed Filtering Method(s) in Terms of F-Score on Reuters-21578 Dataset.

V. Conclusion and Future Directions

Filter-based Feature selection is one of the dimensional reduction techniques, and it is an important preprocessing step of any text classification problem. Selecting the most informative features is one of the main problems faced in building a robust classifier due to performance degradation and time complexity. Reducing the dimension of features by removing irrelevant and noise features as well as retaining the relevant features will significantly reduce the classifiers' computational complexity. There have been a quiet number of works done in the literature to address this problem. The common filtering methods select features by considering a single theoretical approach. Recently, a hybrid approach that combines multiple filtering methods based on different theoretical approach receives more attention. These methods produce a discrepancy in the result that makes combining features subsets produced into a single subset and selecting significant features a difficult task. In this paper, we propose a novel Bi-strategy fileting approach that uses the combined scores of IG and *t*-test to produce refined features subsets by setting a new threshold. The method filters out common features with low V-scores from the considered subsets of features without sacrificing classifier's performance. First, two subsets with high ranked features based on IG and *t*-test are produced. This is done by defining the initial threshold K_1. Then the method identified a feature with minimum IG and t-test scores that are present in both subsets produced and compute their V-scores. The minimum V-score is set as the new threshold K_2, and it is used to further filter out insignificant features from the IG and t-test subsets.

In order to validate the performance of the proposed method, the study presents a comparison based on accuracy and F-score of the filtering approach with that of benchmark methods include IG, t-test (TS), and existing hybrid subsets merging approaches include Union (UA), Intersection (IA) and V-score (VS) using five classification algorithms. The experiment is conducted using three different text datasets, 20Newsgroups, NewsCategoty, and Reutres-21578. Results in Fig. 2 show that our filter method produces a subset with features that is higher than that of IA in number but smaller than that of IG, TS, UA, and VS. It is the fact that our method ignored irrelevant and noisy features and at the same time retained much more informative features, unlike IA. Further experiment results showed that with the small size of features subset produced, our approach achieved a significant improvement in terms of accuracy and F-score of the classifiers used at the cost of a minimum running time. Lastly, a conclusion is reached that the proposed approach achieved a competitive performance even though it does not always give the highest result in most cases, but the results are still acceptable.

In future work, there is a need to investigate the following task: (1) To develop and in-cooperate a feature hashing method as the next step to our method that will consider the correlation between features. (2) To develop a method that has the capability to automatically determine optimal threshold parameter(s) between significant and non-significant features without any domain expert involvement.

Acknowledgment

The authors would like to thank the Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia for the contribution and support toward this research.

References

[1] M. F. Karaca and S. Bayir, "Examining the impact of feature selection methods on text classification" *Int. J. of Adv. Comput. Sci. and Applications*, vol. 8, no. 12, 2017.

[2] F. Salo, A. B. Nassif, and A. Essex, "Dimensionality reduction with IG-PCA and ensemble classifier for network intrusion detection," *Comput.*

Networks, vol. 148, pp. 164–175, 2019, doi: 10.1016/j.comnet.2018.11.010.

[3] M. Shervin, K. Nal, C. Erik, N. Narjes, C. Meysam, and G. Jianfeng, "Deep learning text based classification: A comprehensive review," *ACM compt. Surv.*, vol. 3, no. 62, pp. 1-40, 2021.

[4] D. Wang, H. Zhang, R. Lui, and W. Lv, "Feature selection based on term frequency and t-test for text categorization," *Pattern Recognit. Lett.*, vol. 45, no. 1, pp. 1–6, 2013, doi: 10.1016/j.patrec.2014.02.013.

[5] K. Javed and H. A. Babri, "Feature selection based on a normalized difference measure for text classification," *Inf. Process. Manag.*, vol. 53, no. 2, pp. 473–489, 2017, doi: 10.1016/j.ipm.2016.12.004.

[6] N. Khan, M. S. Husain, and M. R. Beg, "Big data classification using evolutionary techniques : A survey," in *Preceedings of IEEE International Conference on Engineering and Technology (ICETECH)*, 2015, no. March, pp. 243–247.

[7] A. Faraz, "An elaboration of text categorization and automatic text classification through mathematical and graphical modelling," *Comput. Sci. Eng. An Int. J.*, vol. 5, no. 2/3, pp. 1–11, 2015, doi: 10.5121/cseij.2015.5301.

[8] J. Luengo, D. García-gil, S. Ramírez-gallego, S. Garcia, and F. Herrera, *Big Data Preprocessing*, 1st ed. Cham: Springer, 2020.

[9] A. K. Nikhath, K. Subrahmanyam, and R. Vasavi, "Building a K-Nearest Neighbor classifier for text categorization," *Int. J. Comput. Sci. Inf. Technol.*, vol. 7, no. 1, pp. 254–256, 2016.

[10] A. M. Mostafa, "An evaluation of sentiment analysis and classification algorithms for arabic textual data," *Int. J. Comput. Appl.*, vol. 158, no. 3, pp. 29–36, 2017.

[11] F. Barigou, "Improving K-nearest Neighbor efficiency for text categorization," *Neural Netw. World*, vol. 1, no. July, pp. 45–65, 2016, doi: 10.14311/NNW.2016.26.003.

[12] Y. Chen, Q. Zhou, L. Wei, and D. Ji-Xiang, "Classification of chinese texts based on recognition of semantic topics," *Cognit. Comput.*, vol. 8, no. 1, pp. 114–124, 2015, doi: 10.1007/s12559-015-9346-8.

[13] L. N. H. Nam and H. B. Quoc, "A combined approach for filter feature selection in document classification," in *In Proceedings of the International Conference on Tools with Artificial Intelligence, ICTAI*, 2016, vol. 2016-Janua, pp. 317–324, doi: 10.1109/ICTAI.2015.56.

[14] R. Janani and S. Vijayarani, "Automated text classification using machine learning and optimisation algorithms," *Soft Comput..*, vol. 25, no. 1, pp. 1129–1145, 2021.

[15] D. S. Guru, M. Suhil, L. N. Raju, and N. V. Kumar, "An alternative framework for univariate filter based feature selection for text categorization," *Pattern Recognit. Lett.*, vol. 103, no. January, pp. 23–31, 2018, doi: 10.1016/j.patrec.2017.12.025.

[16] M. Micheal and L. Tony, "Evaluation of dimensionality reduction techniques: Principal feature analysis in case of text classification problems ," in *Proceeding of the 16th International Conference on Computing and Data Engineering*, 2020, pp. 75-79.

[17] S. Ayesha, M. K. Hanif, and R. Talib, "Overview and comparative study of dimensionality reduction techniques for high dimensional data," *Inf. Fusion*, vol. 59, no. January, pp. 44–58, 2020, doi: 10.1016/j.inffus.2020.01.005.

[18] M. N. Asim, A. Rehman and U. Shoaib, "Accuracy based feature ranking metric for multi-label text classification" *Int. J. of Adv. Comput. Sci. and Appl. (IJACSA)*, vol. 8, no. 10, 2017.

[19] L. N. Nam Hoai and H. B. Quoc, "A comprehensive filter feature selection for improving document classification," in *Proceedings of 29th Pacific Asia Conference on Language, Information and Computation (PACLIC)*, 2015, pp. 169–177.

[20] M. Habib, C. Sun, A. Abbas, and P. Prakash, "Big data reduction methods : A survey," *Springer-Data Sci. Eng.*, vol. 1, pp. 265–284, 2016, doi: 10.1007/s41019-016-0022-0.

[21] L. Mahdieh, M. Parham, A. Fardin, and J. Mahdi, "A novel multivariant filter method for feature selection in text classification problems*", j. of Eng. Aappli. Of Artificial intelli.*, vol. 70, pp. 25-37, 2017.

[22] S. Suthaharan, Machine Learning Models and Algorithms for Big Data Classification, 2016th ed. Greensboro, USA: Springer, 2016.

[23] N. Pilnenskiy and I. Smetannikov, "Feature selection algorithms as one of the python data analytical tools †," *Futur. internet Artic.*, vol. 54, no. 12, pp. 1–14, 2020, doi: 10.3390/fi12030054.

[24] D. Wang, H. Zhang, R. Liu, X. Liu, and J. Wang, "Unsupervised feature selection through Gram – Schmidt orthogonalization — A word co-occurrence perspective," *Neurocomputing*, vol. 173, pp. 845–854, 2016, doi: 10.1016/j.neucom.2015.08.038.

[25] J. Golay and M. Kanevski, "Unsupervised feature selection based on the morisita estimator of intrinsic dimension," *Springer-Knowledge-Based Syst.*, vol. 135, no. Nov, pp. 125–134, 2017, doi: 10.1016/j.knosys.2017.08.009.

[26] K. Ikeuchi, *Computer Vision: A Reference Guide*, 2014th ed., vol. 2. Boston, USA: Springer, 2014.

[27] M. Li, H. Wang, L. Yang, Y. Liang, and Z. Shang, "Fast hybrid dimensionality reduction method for classification based on feature selection and grouped feature extraction," *Expert Syst. Appl.*, vol. 150, no. July, pp. 1–10, 2020, doi: 10.1016/j.eswa.2020.113277.

[28] A. Onan and K. Serar, "A feature selection model based on genetic rank aggregation for text sentiment classification," *J. Inf. Sci.*, vol. 43, no. 1, pp. 25–38, 2015, doi: 10.1177/0165551515613226.

[29] I. M. El-Hasnony, S. I. Barakat, M. Elhoseny, and R. R. Mostafa, "Improved feature selection model for big data analytics," *IEEE Access*, vol. 8, pp. 66989–67004, 2020, doi: 10.1109/ACCESS.2020.2986232.

[30] Z. M. Hira and D. F. Gillies, "A review of feature selection and feature extraction methods applied on micraarray data," *Adv. Bioinformatics*, vol. 2015, no. July, pp. 1–13, 2015, doi: 10.1155/2015.

[31] A. Subasi, Feature Extraction and Dimension Reduction., 1st ed., Elsevier Inc. 2019.

[32] M. Rong, D. Gong, and X. Gao, "Feature selection and its use in big data: challenges, methods, and trends," IEEE Access, vol. 7, pp. 19709–19725, 2019.

[33] A. D. Rahajoe "Forecasting feature selection based on single exponential smoothing using wrapper method*" Int. J. of Adv. Comput. Sci. and Appl. (IJACSA)*, vol. 10, no. 6, pp. 139-145, 2019.

[34] N. Gu, M. Fan, L. Du, and D. Ren, "Efficient sequential feature selection based on adaptive eigenspace model," *Neurocomputing*, vol. 161, pp. 199–209, 2015, doi: 10.1016/j.neucom.2015.02.043.

[35] M. Rong, D. Gong, and X. Gao, "Feature selection and its use in big data: challenges, methods, and trends," *IEEE Access*, vol. 7, pp. 19709–19725, 2019, doi: 10.1109/ACCESS.2019.2894366.

[36] B. Andrea, S. Xudong, B. Bernd, and L. Micheal, "Benchmark for filter methods feature selection in hugh-dimensional classification data," *Compt. Stat. abd data Analy.*, vol. 143,no. 106839, pp. 1-19, 2020.

[37] H. Zhou, S. Han, and Y. Liu, "A novel feature selection approach based on document frequency of segmented term frequency," *IEEE Access*, vol. 6, pp. 53811–53821, 2018, doi: 10.1109/ACCESS.2018.2871109.

[38] W. Xinzheng, G. Bing, S. Yan, Z.Chimin, and D.Xuliang, "Input feature selection method based on feature set equevalance and mutula information gain maximization," *IEEE Access*, vol. 7, pp. 151525–151538, 2019.

[39] B. Nurhayati, E. P. Armanda, and W. L. Kesuma, "Chi-square feature selection effect on Naïve Bayes classifier algorithm performance for sentiment analysis document," in *Proceedings of the 7th International Conference on Cyber and IT Service Management* , 2019, pp. 1–7.

[40] [40] C. Li and X. Jiucheng, "Feature slection with Fisher score followed by the Maximal Clique Centrality algorithm can accurtely identify the hub genes of hepatocellular carcinoma," *Sci. Rep.*, vol. 9, no. 1, pp. 17283, 2019.

[41] M. Jafari, B. Ghavami, and V. Sattari, "A hybrid framework for reverse engineering of robust Gene Regulatory Networks," *Artif. Intell. Med.*, vol. 79, pp. 15–27, 2017, doi: 10.1016/j.artmed.2017.05.004.

[42] N. El Aboudi and L. Benhlima, "A review on wrapper feature selection approaches," in Proceedings of International Conferencr of Engineeering and MIS (ICEMIS), 2016, pp. 1–5.

[43] D. Ö. Şahin and E. Kılıç, "Two new feature selection metrics for text classification," *J. Control. Meas. Electron. Comput. Commun.*, vol. 60, no. 2, pp. 162–171, 2019, doi: 10.1080/00051144.2019.1602293.

[44] A. K. Das, S. Sengupta, and S. Bhattacharyya, "A Group Incremental Feature Selection for Classification using Rough Set Theory based Genetic Algorithm," *Appl. Soft Comput. J.*, vol. 64, no. April, pp. 400–411, 2018, doi: doi.org/10.1016/j.asoc.2018.01.040.

[45] J. Kittler, *Feature selection and extraction.* Elsevier Inc., 2014.

[46] G. Jesus and Q. G. John, "A new multi-objective wrapper method for feature selection-Accuracy and stability analysis for BCI," *Neurocomputing.*, vol. 333, pp. 407-418, 2017.

[47] C. Hou, F. Nie, X. Li, D. Yi, and Y. Wu, "Joint embedding learning and sparse regression: A framework for unsupervised feature selection," *IEEE Trans. Cybern.*, vol. 44, no. 6, pp. 793–804, 2014, doi: 10.1109/TCYB.2013.2272642.

[48] R. Chakraborty and N. R. Pal, "Feature selection using a neural framework with controlled redundancy," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 26, no. 1, pp. 35–50, 2015, doi: 10.1109/TNNLS.2014.2308902.

[49] T. Kari *et al.*, "Hybrid feature selection approach for power transformer fault diagnosis based on support vector machine and genetic algorithm," *IET Gener. Transm. Distrib.*, vol. 12, no. 21, pp. 5672–5680, 2018, doi: 10.1049/iet-gtd.2018.5482.

[50] H. Zou and T. Hastie, "Regularization and variable selection via the elastic net," *J. R. Stat. Soc. Ser. B Stat. Methodol.*, vol. 67, no. 2, pp. 301–320, 2005, doi: 10.1111/j.1467-9868.2005.00527.x.

[51] D. Loanni, "A review on variable selection in regression Analysis," *econometrics*, vol. 6, no. 25, pp. 1-27, 2018.

[52] D. D. Lewis, "Feature selection and feature extraction for text categorization," in *In Proceedings of Speech and Natural Language: workshop held at Harriman*, 1992, pp. 212–217.

[53] D. Mladenic and G. Marko, "Feature selection for unbalanced class distribution and naive bayes.," in *Proceedings of the Sixteenth International Conference on Machine Learning (ICML)*, 1999, no. January, pp. 258–267.

[54] G. Forman, "BNS Feature Scaling : An Improved representation over TF · IDF for SVM text classification," in *ACM 17th Conference on Information and Knowledge Management*, 2008, pp. 1–8.

[55] H. Uguz, "Knowledge-Based Systems A two-stage feature selection method for text categorization by using information gain , principal component analysis and genetic algorithm," vol. 24, pp. 1024–1032, 2011, doi: 10.1016/j.knosys.2011.04.014.

[56] L. T. Vinh, L. Sungyoung, L. Y. Park, and B. J. D'Auriol, "A novel feature selection method based on normalized mutual information," *Appl. Intell.*, vol. 37, pp. 100–120, 2012, doi: 10.1007/s10489-011-0315-y.

[57] B. Azhagusundari and A. S. Thanamani, "Feature selection based on information gain," in *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, 2013, no. 2, pp. 18–21.

[58] N. Hoque, D. K. Bhattacharyya, and J. K. Kalita, "A mutual information-based feature selection method," *Expert Syst. Appl.*, vol. 2014, no. April, pp. 1–15, 2014, doi: 10.1016/j.eswa.2014.04.019.

[59] C. Tsai and Y. Hsiao, "Combining multiple feature selection methods for stock prediction : Union , intersection , and multi-intersection approaches," *Decis. Support Syst.*, vol. 50, no. 1, pp. 258–269, 2010, doi: 10.1016/j.dss.2010.08.028.

[60] K. K. Bharti and P. K. Singh, "Hybrid dimension reduction by integrating feature selection with feature extraction method for text clustering," *Expert Syst. Appl.*, vol. 42, no. 6, pp. 3105–3114, 2015, doi: 10.1016/j.eswa.2014.11.038.

[61] K. D. Rajab, "New Hybrid features selection method : A case study on websites phishing," *Secur. Commun. Networks*, vol. 2017, no. March, pp. 1–10, 2017.

[62] F. Kamalov and F. Thabtah, "A feature selection method based on ranked vector scores of features for classification," *Ann. Data Sci.*, pp. 1–20, 2017, doi: 10.1007/s40745-017-0116-1.

[63] A. Al-thubaity, N. Abanumay, and Z. Mannaa, "The effect of combining different feature selection methods on arabic text classification," in *14th ACIS1 International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing*, 2013, pp. 219–224, doi: 10.1109/SNPD.2013.89.

[64] S. Vora and H. Yang, "A comprehensive study of eleven feature selection algorithms and their impact on text classification," in *proceeding of Computing Conference*, 2017, no. July, pp. 440–449.

[65] D. Apriliani, T. Abidin, E. Sutanta, A. Hamzah, and O. Somantri "Sentiment analysis for assessment of hotel services review using feature selection approach based-on decision tree" *Int. J. of Adv. Comput. Sci. and Appl. (IJACSA)*, vol. 11, no. 4, pp. 240-245, 2020.

[66] F. Thabtah and F. Kamalov, "Phishing detection : A case analysis on classifiers with rules using machine learning," *J. Inf. Knowl. Manag.*, vol. 16, no. 4, pp. 1–16, 2017, doi: 10.1142/S0219649217500344.

[67] N. O. Essied, I. Othman, and A. H. Osman, "A novel feature selection based on one-way ANOVA F-Test for e-mail spam classification," *Res. J. Appl. Sci. Eng. Technol.*, vol. 7, no. 3, pp. 625–638, 2014, doi: 10.19026/rjaset.7.299.

[68] Y. Liu, S. Ju, J. Wang, and C. Su, "A new feature selection method for text classification based on independent feature space search," *Math. Probl. Eng.*, vol. 2020, pp. 1–14, 2020, doi: 10.1155/2020/6076272.

[69] A. O. Adi, and E. Celebi, "Classification of 20 news group with Naïve Bayes classifier," *in Proceedings of 22nd Signal Processing and Commmunications Applications Conference (SIU)*, 2015, pp. 2150–2153.

[70] "Porter Stemming Algorithm (PSA): http://tartarus.org/ martin/PorterStemmer/ last access: October," 2019.

[71] R. Ansari, M. Salwani, Z. Nor, and S. Faezeh, "Stemming text-based weg page classification using machine learning algorithm: A comparison," Int. *J. of Adv. Conp. Sci. and Appl.*, vol. 11, no. 1, pp. 570–576, 2020.

[72] K. M. M. Rajashekharaiah, S. S. Chikkalli, P. K. Kumbar, and P. S. Babu, "Unified framework of dimensionality reduction and text categorisation," *Int. J. Eng. Techology*, vol. 7, no. November 2018, pp. 648–654, 2018, doi: 10.14419/ijet.v7i3.29.21397.

[73] K. Kowsari, K. J. Meimandi, M. Heidarysafa, S. Mendu, L. Barnes, and D. Brown, "Text classification algorithms: A survey," *Information.*, vol. 10, no. 4, pp. 1–68, 2019, doi: 10.3390/info10040150.

[74] I. Zahidul, L. X. Jixue, L. Jiuyong, L.lin, and K. Wei, "A sementics aware random forest for text classification," *in Proceedings of the 28th ACM International Conference on Information and Knowledge Management.*, 2019, pp. 1061-1070.

[75] N. Jothi, W. Husain, A. N. Rashid Abdul, and S. M. Syed-mohamad, "Feature selection method using genetic algorithm for medical dataset," *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 9, no. 6, pp. 1907–1912, 2019.

[76] Z. Xiaoming, C. Wenhan, L.Zhoujun, L. Chunyang, and L.Rui, "Multi-modal kernel ridge regression for social image classification", *Applied Soft Cmmpt.*, vol. 67, pp. 117–125, 2018.