Comparative Analysis of Integrating Multiple Filter-Based Feature Selection Methods Using Vector Magnitude Score on Text Classification

Abubakar Ado, Noor Azah Samsudin, Mustafa Mat Deris
Faculty of Computer Science and Information Technology
Universiti Tun Hussein Onn Malaysia
Batu Pahat 86400, Malaysia
abbakarrgg@gmail.com, azah@uthm.edu.my, mmustafa@uthm.edu.my

Abdulkadir Abubakar Bichi
Computer Science Department
Universiti Teknologi Malaysia
Johor, Malaysia
engraabubakar@gmail.com

Aliyu Ahmed
Computer Science Department
Bauchi State University
Gadau, Nigeria
ahmedaliyu8513@gmail.com

Abstract

High-dimensionality is one of the major problems that arise in text classification task. Usually, dimensional reduction techniques are used to reduce the feature dimensions to a minimum number without not or much affecting the classifiers' performance. Among the techniques, filter-based is the widely used, aiming to select the informative features from the original features set. The filter methods proposed in literature falls into critical problem of being not much effective with respect to some datasets or classifiers. To overcome such issue, a number of works were presented combining different multiple filter methods. This approach improves classifiers' performance by maximizing the advantage of one method and minimizing the disadvantage of the other. In this paper, we studied the impact of combining multiple FS methods, comprising MI, Chi2, and t-test, on a text classification problem. V-score is adapted to combine and ranked the features produced by the chosen FS methods. Experiment is conducted on movie reviews dataset and classification accuracy is reported using NB and SVM. Both the methods were evaluated based on TFIDF and Count_vector feature representations. Experimental results demonstrate minor improvement in performance by combining two filter methods and no significant improvement by combining the three methods.

Keywords

Feature Selection, Filter-Based, Feature Representation, Dimensionality Reduction, and Classifier

1. Introduction

In today's text data Analysis, high-dimensionality has become the main problem faced by machine learning models. The massive number of features generated by textual data introduces the so-called "curse of dimensionality", and this aggressively increases the computational complexity of machine learning models (Habib et al., 2016; Shepitsen et al., 2008; Suthaharan, 2016). For instance, in documents classification, the dataset contains tens to hundreds of thousands features. Feeding such dataset into our state of art learning models or even most advance models is tedious and sometimes infeasible. However, many kind of these datasets might contain features that are irrelevant, redundant or noise that course serious problem to the classification task (Vora & Yang, 2017). Dimensionality reduction is the

current research direction that helps overcome the problems mentioned earlier (Ikeuchi, 2014; Krishnan et al., 2019). The focus point behind dimensionality reduction is reducing the number of original features set to a smaller informative features subset so that it can be easily handled by the prediction model (Caragea et al., 2012; Juvonen et al., 2015). Feature Selection (FS) (Rong et al., 2019; Sharif et al., 2018) is one of the main techniques of dimensional reduction, which is more applicable to a large dataset. The technique selects from the original feature set, the most informative features subset that contributes to the classification task without sacrificing the performance of the classification models (Al-thubaity et al., 2013; El-Hasnony et al., 2020; Onan & Serar, 2015). Application domains that contain a large number of samples with a large number of features extensively benefit via FS techniques. Tasks such as Sentiment Analysis, Spam Filtering, Recommender System, Text Mining and Documents Categorization require effective FS methods to preserve or improve classification models' performance with respect to lower running time (M. Rajab & Wang, 2020).

Generally, FS methods (Forman, 2003) are broadly grouped into filter methods, wrapper methods, and embedded methods (Gu et al., 2015; Li et al., 2020; Rong et al., 2019). Filter methods (Bermejo et al., 2013) are independent that they do not interact with classifier when constructing an informative feature 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 (Li et al., 2020). They select features subset by considering the usefulness of a feature according to evaluation metrics(Li et al., 2020; Onan & Serar, 2015; Rong et al., 2019; Zhou et al., 2018). Filter methods usually have good computational efficiency but affect classification accuracy to some point. Information Gain (Quinlan, 1986), Chi-Square (H. Liu & Setiono, 1995), Fisher Score (Longford, 1987), ReliefF (Jafari et al., 2017), t-test (Wang et al., 2013) 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 (Li et al., 2020; Nam & Quoc, 2016; Rong et al., 2019). They evaluate a particular feature subset by training and testing a given classifier, and they are personalized to a particular classifier (Saeys et al., 2007). These methods have bad computational efficiency but produce high classification accuracy. These methods are not usually favored in text classification task (Şahin & Kılıç, 2019). Heuristic Search Algorithms (HSA) and Sequential Selection Algorithms (SSA) (Das et al., 2018; Kittler, 2014; Kohavi & John, 1997) 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 (Rong et al., 2019)(Das et al., 2018)(Hou et al., 2014). Like wrapper methods, embedded methods frequently interact with the classifier but have computational efficiency better than wrapper methods, and are also personalized to a specific classifier (Şahin & Kılıç, 2019). Selection-Perceptron (FS-P)(Chakraborty & Pal, 2015), Support Vector Machines (SVM-RFE) (Kari et al., 2018), Lasso (L1) and Elastic Net (L1+L2) based models (Tibshirani, 1996; Zou & Hastie, 2005) are some few examples of embedded based methods. This study is focusing only on filter-based FS methods.

Many studies have analyzed numerous filter-based FS methods, and most of these investigation works analyze the impact of different filter methods to classification models by considering a single strategy. However, very few studies investigate the challenges and impact of combining multiple strategies. For example, result discrepancy is among the significant challenges associated with combining multiple methods (Kamalov & Thabtah, 2017), making it hard for two distinct filter methods to select the same set or very similar set of features (Vora & Yang, 2017). The feature similarities of combining two filter methods may be lower or higher than those of other different combinations. In (Vora & Yang, 2017), it has shown that the average similarity score obtained with IG and Chi2 falls between (0.93 - 1.00) while for CFS and FCBF falls between (0.65 - 0.32). Thus the later combination selects more identical features. This issue seriously impacts combining multiple filter methods, and wrong combination may result in the degradation performance of classification models.

This paper investigates the impact of combining multiple filter-based FS methods with regard to text classification upon two widely used classifiers. The study wishes to answer the following research questions: Will multiple combinations of different filter methods, have an impact on classification performance? What is the impact of combining two versus three filtering FS methods using vector magnitude score (V-score) approach with respect to text classification?

The remaining body of this paper is systematically partitioned as follows: In Section 2, related works are presented. Selected Filter-based feature selection methods chosen for the study are briefly discussed in section 3. Properties of the datasets used and experimental set up are devoted to section 4. Experimental results and analysis 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.

2. Literature Review

As mentioned earlier, Feature selection is a dimensionality reduction techniques that assist in evaluating the significance of a feature. Increasing the optimality of features subset is the main idea behind combining different FS methods. Two main criteria administer combination of FS, the strategy utilized in selecting features for the combination process, and the approach of combining the selected features sets (Al-thubaity et al., 2013). Recently, few studies have proposed approaches for merging features subsets and filtering optimal features based on combining multiple FS methods such as Union(OR) approach (Tsai & Hsiao, 2010), Intersection (AND) (Tsai & Hsiao, 2010; Uguz, 2011), Modified Union approach (Bharti & Singh, 2015), and V-score (Kamalov & Thabtah, 2017; K. D. Rajab, 2017) approach. This study is based on vector magnitude score (V-score). The approach combines the scores of each feature computed by both filter methods employed and later selects the most informative features by considering a defined threshold.

The effectiveness of combining multiple FS methods based on predictive models performance has been investigated in many studies. Most of the studies reported performance improvement in accuracy and processing time when multiple distinct FS methods are combined. (Tsai & Hsiao, 2010) hybridize multiple methods for dimensional reduction to figure out more informative features for stock prices prediction task. In their study, (Onan & Serar, 2015) essembled seven different FS methods and employed Genetic Algorithm (GA) to marge the independent feature sequences into a single comprehensive sequence in text sentiment Analysis problem. An intermediary method of Union (OR) approach and Intersection (AND) approach named modified union is presented by (Bharti & Singh, 2015). The authors in (Nam & Quoc, 2016) proposed a hybrid filtering method for selecting optimal features subsets in large scale textual problem by integrating cluster-based and frequent-based approach, termed FCFS. 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 (K. D. Rajab, 2017). Study presented in (Kamalov & Thabtah, 2017) proposed a method that selects optimal features from sets with ranking features produced by three different ranking strategies. Forman (Forman, 2003) empirically studied and compared twelve different evaluation metrics for feature selection on a text classification problem. They finally revealed that BNS with IG has the minimum correlated failure so as mark best backup choice. (Molina et al., 2002) present a comprehensive study that reveals the behaviour of various FS methods based on the criteria of irrelevance, redundancy and relevance. The impact of integrating five methods for FS was investigated by (Al-thubaity et al., 2013). In another new multi-stage approach, (Li et al., 2020) consider the application of union approach on the lowest rank feature subset produce by Fisher score and IG methods in the first phase of their method. 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 & Yang, 2017) present a comparative study on ten different filtering methods: Fisher Score, Chi-square, Gini Index, Laplacian Score, IG, mRmR, and 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 feature been selected.

3. Feature Selection Methods

As we discussed earlier, FS methods are usually applied to high dimensional datasets to remove irrelevant, redundant, and noise features. The methods efficiently reduce the dimension of features set, improve the classification performance and at the same time, reduce processing time. In this section, we briefly discuss the chosen benchmark Filter-based FS methods, comprising Mutual Information (IG), Chi-Square (Chi2), and Student Statistical Test (t-test).

3.1. Mutual Information (MI)

The Mutual Information (MI) (Haris B. & Revanasidappa M., 2017; Lefkovits & LefKvotis, 2017) is an information theory-based commonly used in statistics to model word association (Haris B. & Revanasidappa M., 2017). MI measures the amount of information presence or absence of a feature and its contribution in making the appropriate classification prediction on labelled classes. The method computes and assigns a score to each feature by considering the variation between the entropy obtained based on the presence or absence of feature/term in a class (Zhou et al.,

2018). High MI score indicates the discriminating capability of a feature and ranked top. Given a random variable *X* and a specific event x_i, let $P(x_i)$ denotes the probability of an event (x_i). Mathematically, the entropy of discrete random variable *X* is formulated as:

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

The general formula for calculating MI of a given feature *t* is formally given as:

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

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:

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

In the 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 $\mathrm{I(t,c)} = 0$. MI 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.

3.2. Chi-Square (Chi2)

Chi-Square (Chi2) (Haris B. & Revanasidappa M., 2017; H. Liu & Setiono, 1995) is a statistical-based and important non-parametric test method used to Compare more than two attributes for a randomly chosen data (Şahin & Kılıç, 2019). It is commonly known as independence test, and the method is applied to test if the concurrence of a particular term and a particular category are not dependent. Chi2 generates a value that reveals the relationship between a term and category during feature filtering process in text classification. The Chi2 score of a give term t_k for a category c_i is calculated as:

$$Chi - square(t_k\ c_i) = \frac{NP(t_k)^2\big(P(c_i|t_k) - P(c_i)\big)^2}{P(t_k)\big(1 - P(t_k)\big)P(c_i)\big(1 + P(c_i)\big)} \tag{4}$$

Where $P(t_k)$ is the probability of a document that contains term t_k, $P(c_i)$ is the probability of documents that belongs to class c_i; $P(c_i|t_k)$ is the conditional probabilities of a document that belongs to c_i; given that it contained t_k. If the two events are dependent, then the term occurrence makes the class occurrence less likely (or more likely), so it should be considered as contributing feature. Thus the importance for each feature is estimated and rank them according to their score. If a feature is near to more classes, then the score of that feature will be higher. Furthermore, this score can be generalized over all classes in two approaches. The first approach calculates the weighted average score for all class, while the second approach is to select the maximum score between all classes (Haris B. & Revanasidappa M., 2017).

$$Chi2_{avg}(t) = \sum_{i=1}^{m} P_r\,(c_i) Chi2(t, c_i) \tag{5}$$

$$Chi2_{max}(t) = \max_i\{Chi2(t, c_i)\} \tag{6}$$

3.3. Student Statistical Test (*t-test*)

Student Statistical Test (*t*-test) (Wang et al., 2014) 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 (Essied et al., 2014; Wang et al., 2013, 2014). Presently, t-test is widely used as an evaluation function to select significant features that contribute to classifying instances. The method computes the t-score of each feature by measuring the term's distinct distributions in relevant class and documents collection (Y. Liu et al., 2020). The series of formulations for calculating t-test is given as:

$$t - test(t_i, c_k) = \frac{\left|\overline{tf_{kl}} - \overline{tf_l}\right|}{m_k \times s_i} \tag{7}$$

$$S_i^2 = \frac{1}{N-K}\sum_{k=1}^{k}\sum_{j \in C_k}(tf_{ij} - \overline{tf_{kl}})^2 \tag{8}$$

$$m_k = \sqrt{\frac{1}{N_k} - \frac{1}{N}} \tag{9}$$

Each class's specific scores obtained from equation (7) are combined to find the final score.

$$t - test_{avg}(t_i) = \sum_{k-1}^{k} t - test(t_i, C_k) \tag{10}$$

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} donates the average *TF* of term t_i in k^{th} category, tf_i denotes average *TF* of term t_i in the corpus. *N* denotes the total number of documents in the collection. When t-test score is less than a defined threshold, it indicates that the feature has lower discrimination ability; otherwise, the feature will contribute to the classifying instances and will be selected.

4. Experiment and Dataset

In this section, a summary of the datasets used and the implantation process adapted are briefly explained. Also, the experimental settings are briefly explained. Lastly, the section ends with a brief discussion on Classifiers selected for the experiment.

4.1. Dataset

The *Movie Reviews* dataset is a collection of movie review extracted from the imdb website. The dataset was initially made available in 2002, later in 2004, an updated version (referred to as "*v2.0*") was released. The dataset contains features of frequently used nouns, verbs, adverbs and adjectives. It comprises of 2000 instances, 1,000 negative and 1,000 positive movie reviews. In this study, we used the public available *movie reviews* dataset from two corresponding corpora included in the Python NLTK toolkit for Natural Language Processing. We partitioned it into 70% for training and 30% for testing. The summarized properties of the dataset can be found in table1.

4.2. Experiment Settings

In this experiment, the effect of combining multiple filter-based FS methods on text data are studied. Three existing benchmark feature filtering methods are considered for the comparative analysis, namely IG, Chi2, and t-test. The features considered for the training are selected based on feature ranking where individual features from the chosen benchmark FS methods are combined using Vector Magnitude Score (V-score) approach. All features are ranked based on V-score, and all those high ranked features above a defined threshold are used for classifying instances. Thresholds defined are listed in table 1. Firstly, the classification accuracy based on individual benchmark filter methods is evaluated. Furthermore, these methods are hybridized with each other using the above-mentioned combining approach, and their impact on the classification accuracy is evaluated.

4.3. Classifiers

Two well-known classifiers are used for the analysis evaluation purpose in this study, including Support Vector Machine (SVM) (Suthaharan, 2016) and Naïve Bayes (NB) (Kowsari et al., 2019). Both the classifiers are selected based on their performance in the context of text classification problem.

SVM is a binary classifier proposed by Vapnik and Chervonenkis, formulated from statistical learning theory (Unnikrishnan et al., 2017). The model marks the maximum channel between two categories known as a hyperplane [B5] and separates the data samples into different categories (Estévez et al., 2019). Regarding text classification, let $(x_1, x_2 \ldots . x_l)$ be training samples belonging to a class X, where X is a dense subset of R^N (Suthaharan, 2016). Then SVM classifier is modeled as follows:

$$\min \frac{1}{2}\|w\|^2 + \frac{1}{vl}\sum_{i=1}^{l}\zeta_i - p \tag{11}$$

Subject to:

$$(w.\Phi(x_i)) \geq p - \zeta_i, i = 1,2, \ldots l, \qquad \zeta \geq 0 \tag{12}$$

Decision function is expressed as the following equation if and only if w and p solve the problem.

$$f(x) = sign((w.\Phi(x)) - p \tag{13}$$

Naïve Bayes (NB) is also a supervised classifier method which is theoretically established based on Bayes theorem, and it was proposed by Thomas Bayes (Suthaharan, 2016)(Kowsari et al., 2019). It uses 'naïve' independence between every features or attributes pair (Al-thubaity et al., 2013). The model integrates maximum a posteriori (MAP) decision rule together with the Bayes model (Al-thubaity et al., 2013). Regarding text classification, let n be the number of documents that fit into K categories, where $K\epsilon \{c_1, c_{2,}, \ldots c_k\}$, the predicted category's output is given as c ∈ C, and d and c denotes documents and categories. Then NB is modeled as follows:

$$P(d|c) = \frac{P(d|c)P(c)}{P(d)} \tag{14}$$

$$C_{MAP} = \arg\max_{c\epsilon C} P(d|c)\, P(c) = \arg\max_{c\epsilon C} P(x_1, x_2, \ldots, x_n)\, p(c) \tag{15}$$

4.4. Software Implementation

In this study, all the implementations for the experiments are conducted on Python (V3.8.2) environment, which is installed on a computer with Windows 8 (OS). Other minimum required conditions for the experiments include Intel(R) Core™ i5 processor4300m@2.60GHz/8GRAM/64 GB/Windows 8. Python's implementation from scikit-learn is used for all the three chosen FS methods and the two classification methods. Default values of most of the parameters associated with the classification methods are retained.

Table 1: Summary of Dataset and Parameters Used in the Experiment

Parameters	Description
Dataset	Movie Reviews
#Samples	2000
#Features	49582
#classes	2
Feature Representations	Count_vector & TFIDF
Thresholds Set	20%, 40%, 60% and 80%

Classifiers Parameters	Default

5. Results and Analysis

Given classification methods, is there any significant impact on their performance by applying different FS methods? Also, will multiple combinations of different filter methods using vector magnitude score (V-score) approach have an impact on classification performance? Furthermore, what is the impact of combining two VS three filtering methods with respect to the selected classifiers? To answer these questions, a series of experiments and evaluations have been conducted. As mentioned earlier in section 3, that this comparative analysis includes three FS methods and two well-known classifiers. For each benchmark FS methods and combination of them, the top-ranked features above a defined threshold(s) will be evaluated using each of the chosen classifiers.

5.2 Graphical Results

Shown in Figure. 1 and Figure. 2 are classification accuracy of the two classifiers (NB and SVM) been evaluated on *Movie Reviews* dataset by applying single benchmark FS methods (MI, Chi2, and t-test*)* and multiple combinations of them *(MI+Chi2, MI+t-test, Chi2+t-test, and MI+Chi2+t-test)*, respectively. Closely looking at the classification accuracy presented based on NB in figure1, we can clearly see that, *MI+Chi2* recorded the highest accuracy of 82.33% using TFIDF followed by *MI+t-test* with 81.33%, while *Chi2+t-test* recorded the lowest accuracy. Likewise, for Count_vector feature representation, *MI+Chi2* also recorded the highest accuracy of 80.11% followed by *MI+t-test* with 79.77% while *MI* recorded the lowest accuracy. The highest accuracy was 83.83% and 87.24, which is also recorded by *MI+Chi2* while the lowest accuracy was recorded by *t-test* using both the two feature representations. By considering only the three benchmark FS methods, The *Chi2* outperformed the other two methods for Count_vector while *t-test* superseded using TFIDF. On the other hand, *MI* outperformed the other two benchmark methods for both the feature presentations with small significance enhancements, as shown in figure 2. From the summary of the below results, we can conclude that *MI* works better with SVM for both feature representations while *Chi2* works better with NB if the features are represented using TFIDF otherwise *t-test* works better. We also noticed a significant impact in combining two different FS methods, especially when *MI* is combined with *Chi2* despite they operate based on different theoretical strategy. Moreover, there is no positive impact in combining three different FS methods as depicted from the results. Generally, all the FS methods comparatively performed better with both the classifiers when TFIDF feature representation is used.

Figures 3 and 4 show the results obtained from combinations of multiple FS methods for FTIDF on all the number of features and certain defined thresholds as listed in table 1. We consider only TFIDF due to space limit, and because it

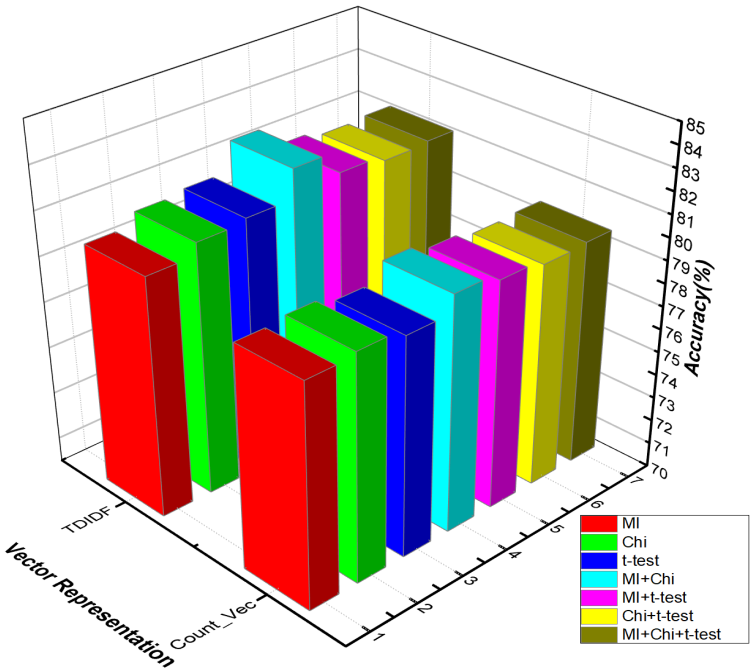

Figure 1. Classification Accuracy Based on NB

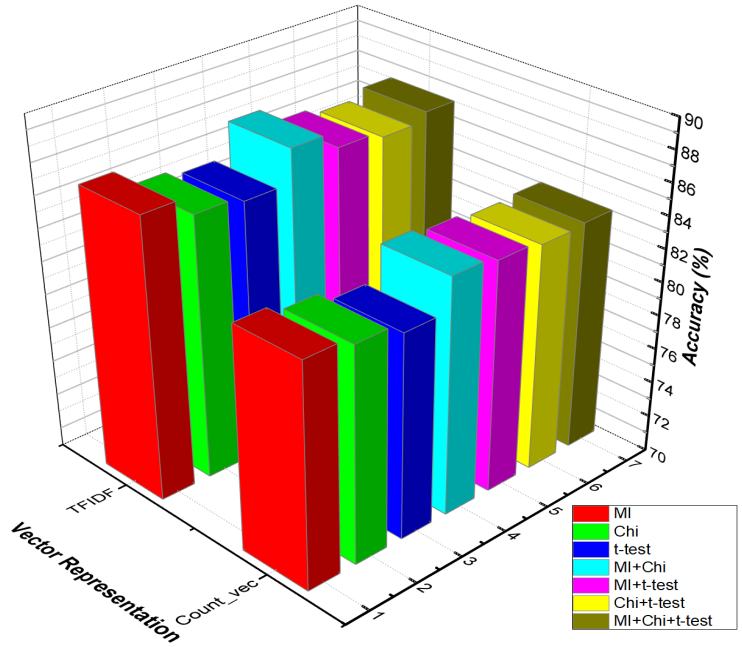

Figure 2. Classification Accuracy Based on SVM

outperformed Count_vectore generally. From Figure 3 it can be seen the classification accuracy recorded using NB. When the number of features selected is 80% of the total number of features, the classification accuracy of *MI+Chi2* is better than the other combination methods upon all classifiers. At 20% threshold, all the methods recorded the lowest accuracy with a combination of MI and Chi2 recorded the highest. This is because most of the informative features are eliminated from the features subset by the methods at that defined threshold. The maximum accuracy is observed with all the methods when all the features are used (100%). *MI+Chi2* outperformed all the other methods

across all defined thresholds except at 40% where no accuracy improvement is achieved. Most of the methods show a significant rear improvement between 60% and 80% with *Chi2+ t-test* achieved the most remarkable improvement of about 0.77% accuracy. This shows most of the features eliminated between the thresholds are non-relevant or noise

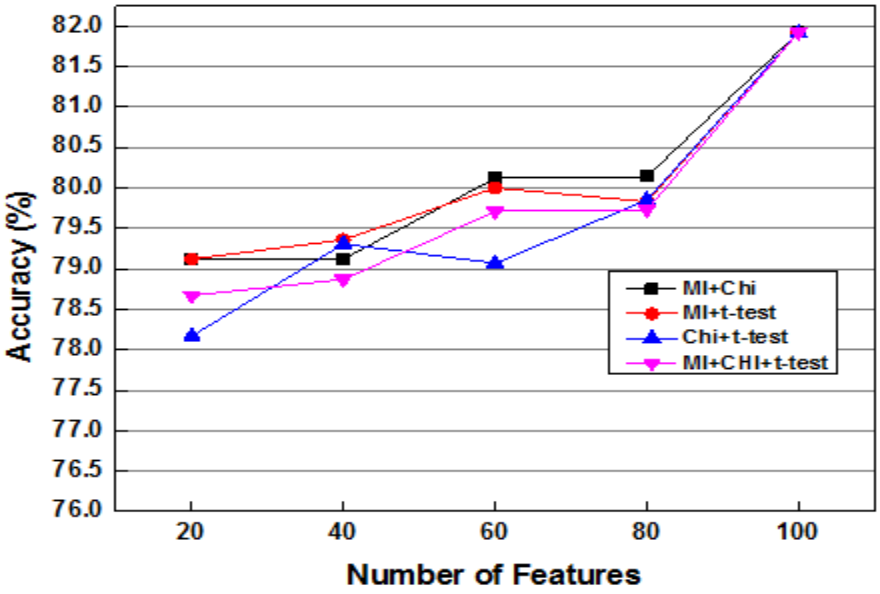

Figure 3. Classification Accuracy of NB on Different Defined Thresholds

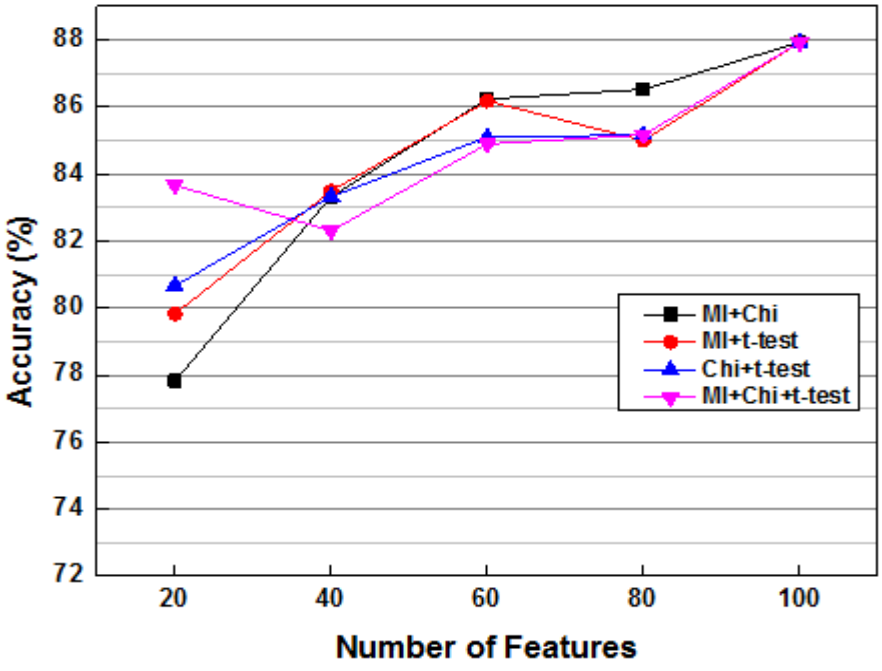

Figure 4. Classification Accuracy of SVM on Different Defined Thresholds

In the other hand, figure 4 reported accuracy results obtained based on SVM. As we can see with the lowest number of features (20%), *MI+Chi2* achieved the lowest accuracy, but it outperformed the other methods when the features increase to 80%. Moreover, *MI+Chi2+t-test* recorded the highest accuracy at the 20% threshold, which drops by 1.38% when the number of features increased to 40%. Most of the features selected at that particular threshold are non-relevant or noise. Like previous evaluation results, all the methods also shows no significant improvement

between 60% and 80% thresholds. Generally, we noticed that with a lower number of features, the ratio of informative features retained by *MI+Chi2* is lower than that of the other methods which turn to be higher when the number of features increases.

6. Conclusion and Future Work

This paper presents a comparative study to investigate the impact of integrating multiple filter-based FS methods and their performance upon two different feature representations. Moreover, the investigation also includes how these FS methods perform with two common distinct classifiers with respect to text classification. Three benchmark FS methods were considered in this study including MI, Chi2, and t-test, these methods were integrated with each other (MI+Chi2, MI+t-test, Chi2+t-test, and MI+Chi2+t-test) using V-score approach in which four defined thresholds of 20%, 40%, 60%, and 80% are set to select the qualified features. TFIDF and Count_vector are used for feature representations. An experiment is conducted using movie reviews dataset. Two chosen classifiers, namely NB and SVM were used to report the classification accuracy. Results from a series of experiments showed a little significant improvement when two FS methods were integrated, with no improvement when the three methods were integrated. The combination of MI and Chi2 achieved the maximum improvement of 0.44% accuracy when couple with NB, and 0.88% when coupled with SVM. Generally, out of the two feature representations, all the methods comparatively performed better using TFIDF than Count_vector. On the other hand, another experiments that consider only multiple FS methods were conducted by reconsidering the four defined thresholds and the whole features set. Results showed that MI+Chi2 outperformed the other methods for both thresholds except 40% with NB classifier and 20% with SVM. Finally, results also reveal that it is worthwhile to integrate MI and Chi2 FS methods in text classification problems when V-score approach is adapted.

In future work, we plan to investigate the following task: (1) To investigate the impact of combining other FS methods based on V-score approach. (2) To propose another combining approach that will improve classifier's performance when t-test is combined with other methods.

7. Acknowledgements

The authors would like to thank the Ministry of Higher Education, Malaysia for supporting this research under Fundamental Research Grant Scheme Vot K213 (FRGS/1/2019/ICT02/UTHM/02/2) and Universiti Tun Hussein Onn Malaysia for Multidisciplinary Research, Vot H511.

References

Al-thubaity, A., Abanumay, N., & Mannaa, Z. (2013). The Effect of Combining Different Feature Selection Methods on Arabic Text Classification. *2013 14th ACIS1 International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing*, 219–224. https://doi.org/10.1109/SNPD.2013.89

Bermejo, P., Jos´e, A. G., & Jos´e, M. P. (2013). Speeding Up Incremental Wrapper Feature Subset Selection with Naive Bayes Classifier. *Knowledge-Based Systems*, *54*, 140–147.

Bharti, K. K., & Singh, P. K. (2015). Hybrid dimension reduction by integrating feature selection with feature extraction method for text clustering. *Expert Systems with Applications*, *42*(6), 3105–3114. https://doi.org/10.1016/j.eswa.2014.11.038

Caragea, C., Silvescu, A., & Mitra, P. (2012). Combining Hashing and Abstraction in Sparse High Dimensional Feature Spaces. *Proceedings of the 26th AAAI National Conference on Artificial Intelligence*, 3–9.

Chakraborty, R., & Pal, N. R. (2015). Feature selection using a neural framework with controlled redundancy. *IEEE Transactions on Neural Networks and Learning Systems*, *26*(1), 35–50. https://doi.org/10.1109/TNNLS.2014.2308902

Das, A. K., Sengupta, S., & Bhattacharyya, S. (2018). A Group Incremental Feature Selection for Classification using Rough Set Theory based Genetic Algorithm. *Applied Soft Computing Journal*, *64*(April), 400–411. https://doi.org/doi.org/10.1016/j.asoc.2018.01.040

El-Hasnony, I. M., Barakat, S. I., Elhoseny, M., & Mostafa, R. R. (2020). Improved Feature Selection Model for Big Data Analytics. *IEEE Access*, *8*, 66989–67004. https://doi.org/10.1109/ACCESS.2020.2986232

Essied, N. O., Othman, I., & Osman, A. H. (2014). A Novel Feature Selection Based on One-Way ANOVA F-Test for E-Mail Spam Classification. *Research Journal of Applied Sciences, Engineering and Technology*, *7*(3), 625–638. https://doi.org/10.19026/rjaset.7.299

Estévez, P. A., Príncipe, J. C., & Zegers, P. (2019). Advances in Intelligent Systems and Computing: Preface. In *Advances in Intelligent Systems and Computing* (Vol. 1039). Springer. https://doi.org/10.1007/978-3-642-

35230-0

Forman, G. (2003). An Extensive Empirical Study of Feature Selection Metrics for Text Classification. *Journal of Machine Learning Research*, *3*(3003), 1289–1305.

Gu, N., Fan, M., Du, L., & Ren, D. (2015). Efficient Sequential Feature Selection Based on Adaptive Eigenspace Model. *Neurocomputing*, *161*, 199–209. https://doi.org/10.1016/j.neucom.2015.02.043

Habib, M., Sun, C., Abbas, A., & Prakash, P. (2016). Big Data Reduction Methods : A Survey. *Springer-Data Science and Engineering*, *1*, 265–284. https://doi.org/10.1007/s41019-016-0022-0

Haris B., S., & Revanasidappa M., B. (2017). A Comprehensive Survey on various Feature Selection Methods to Categorize Text Documents. *International Journal of Computer Applications*, *164*(8), 1–7. https://doi.org/10.5120/ijca2017913711

Hou, C., Nie, F., Li, X., Yi, D., & Wu, Y. (2014). Joint embedding learning and sparse regression: A framework for unsupervised feature selection. *IEEE Transactions on Cybernetics*, *44*(6), 793–804. https://doi.org/10.1109/TCYB.2013.2272642

Ikeuchi, K. (2014). *Computer Vision: A reference Guide* (K. Ikeuchi (ed.); 2014th ed., Vol. 2). Springer.

Jafari, M., Ghavami, B., & Sattari, V. (2017). A hybrid framework for reverse engineering of robust Gene Regulatory Networks. *Artificial Intelligence in Medicine*, *79*, 15–27. https://doi.org/10.1016/j.artmed.2017.05.004

Juvonen, A., Sipola, T., & Hämäläinen, T. (2015). Online anomaly detection using dimensionality reduction techniques for HTTP log analysis. *Computer Networks*, *91*, 46–56. https://doi.org/10.1016/j.comnet.2015.07.019

Kamalov, F., & Thabtah, F. (2017). A Feature Selection Method Based on Ranked Vector Scores of Features for Classification. *Annals of Data Science*, 1–20. https://doi.org/10.1007/s40745-017-0116-1

Kari, T., Gao, W., Zhao, D., Abiderexiti, K., Mo, W., Wang, Y., & Luan, L. (2018). Hybrid feature selection approach for power transformer fault diagnosis based on support vector machine and genetic algorithm. *IET Generation, Transmission and Distribution*, *12*(21), 5672–5680. https://doi.org/10.1049/iet-gtd.2018.5482

Kittler, J. (2014). Feature selection and extraction. In *Handbook of pattern recognition and image processing*. Elsevier Inc. https://doi.org/10.1016/b978-0-12-409545-8.00002-9

Kohavi, R., & John, G. H. (1997). Wrappers for feature subset selection. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *97*(1–2), 273–324. https://doi.org/10.1016/S0004-3702(97)00043-X

Kowsari, K., Meimandi, K. J., Heidarysafa, M., Mendu, S., Barnes, L., & Brown, D. (2019). Text classification algorithms: A survey. *Information (Switzerland)*, *10*(4), 1–68. https://doi.org/10.3390/info10040150

Krishnan, R., Samaranayake, V. A., & Jagannathan, S. (2019). A Hierarchical Dimension Reduction Approach for Big Data with Application to Fault Diagnostics. *Journal of Big Data Research*, *18*, 100121. https://doi.org/10.1016/j.bdr.2019.100121

Lefkovits, S., & LefKvotis, L. (2017). Gabor Feature Selection Based on Information Gain. *10th International Conference Interdisplinarity in Engineering, INTER-ENG 2016*, *181*, 892–898. https://doi.org/10.1016/j.proeng.2017.02.482

Li, M., Wang, H., Yang, L., Liang, Y., & Shang, Z. (2020). Fast hybrid dimensionality reduction method for classification based on feature selection and grouped feature extraction. *Expert Systems With Applications*, *150*(July), 1–10. https://doi.org/10.1016/j.eswa.2020.113277

Liu, H., & Setiono, R. (1995). Chi2 : Feature Selection and Discretization of Numeric Attributes. *Proceedings of the IEEE 7th International Conference on Tools with Arti?Cial Intelligence*, 2–5. https://doi.org/10.1109/TAI.1995.479783

Liu, Y., Ju, S., Wang, J., & Su, C. (2020). A New Feature Selection Method for Text Classification Based on Independent Feature Space Search. *Mathematical Problems in Engineering*, *2020*, 1–14. https://doi.org/10.1155/2020/6076272

Longford, N. T. (1987). A fast scoring algorithm for maximum likelihood estimation in unbalanced mixed models with nested random effects. *Biometrika*, *74*(4), 817–827. https://doi.org/10.1093/biomet/74.4.817

Molina, L. C., Belanche, L., & Nebot, A. (2002). Feature Selection Algorithms : A Survey and Experimental Evaluation. *IEEE International Conference on Data Mining, 2002. Proceedings.*, 306–313. https://doi.org/10.1109/ICDM.2002.1183917

Nam, L. N. H., & Quoc, H. B. (2016). A combined approach for filter feature selection in document classification. *In Proceedings of the International Conference on Tools with Artificial Intelligence, ICTAI*, *2016-Janua*, 317–324. https://doi.org/10.1109/ICTAI.2015.56

Onan, A., & Serar, K. (2015). A feature selection model based on genetic rank aggregation for text sentiment classification. *Journal of Information Science*, *43*(1), 25–38. https://doi.org/10.1177/0165551515613226

Quinlan, J. R. (1986). Induction of Decision Trees. *Machine Learning 1:*, 81–106.

Rajab, K. D. (2017). New Hybrid Features Selection Method : A Case Study on Websites Phishing. *Security and Communication Networks*, *2017*(March), 1–10.

Rajab, M., & Wang, D. (2020). Practical Challenges and Recommendations of Filter Methods for Feature Selection. *Journal of Information & Knowledge Management*, *19*(1), 1–15. https://doi.org/10.1142/S0219649220400195

Rong, M., Gong, D., & Gao, X. (2019). Feature Selection and Its Use in Big Data: Challenges, Methods, and Trends. *IEEE Access*, *7*, 19709–19725. https://doi.org/10.1109/ACCESS.2019.2894366

Saeys, Y., Inza, I., & Larranaga, P. (2007). A review of feature selection techniques in bioinformatics. *Bioinformatic*, *23*(19), 2507–2517. https://doi.org/10.1093/bioinformatics/btm344

Şahin, D. Ö., & Kılıç, E. (2019). Two new feature selection metrics for text classification. *Journal for Control, Measurement, Electronics, Computing and Communications*, *60*(2), 162–171. https://doi.org/10.1080/00051144.2019.1602293

Sharif, W., Samsudin, N. A., Deris, M. M., & Khalid, S. K. A. (2018). A Technical Study on Feature Ranking Techniques and Classification Algorithms. *Journal of Engineering and Applied Sciences*, *13*(9), 7074–7080. https://doi.org/10.3923/jeasci.2018.7074.7080

Shepitsen, A., Gemmell, J., Mobasher, B., & Burke, R. (2008). Personalized Recommendation in Social Tagging Systems Using Hierarchical Clustering. *2008 ACM Conference on Recomemender Systems, RecSys'08*, 259–266. https://doi.org/10.1145/1454008

Suthaharan, S. (2016). *Machine Learning Models and Algorithms for Big Data Classification* (2016th ed.). Springer. https://doi.org/10.1007/978-1-4899-7641-3

Tibshirani, R. (1996). Regression Shrinkage and Selection Via the Lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, *58*(1), 267–288. https://doi.org/10.1111/j.2517-6161.1996.tb02080.x

Tsai, C., & Hsiao, Y. (2010). Combining multiple feature selection methods for stock prediction : Union , intersection , and multi-intersection approaches. *Decision Support Systems*, *50*(1), 258–269. https://doi.org/10.1016/j.dss.2010.08.028

Uguz, H. (2011). *Knowledge-Based Systems A two-stage feature selection method for text categorization by using information gain , principal component analysis and genetic algorithm*. *24*, 1024–1032. https://doi.org/10.1016/j.knosys.2011.04.014

Unnikrishnan, A., Narayana, U., & Joseph, S. (2017). Performance Analysis of Various Supervised Algorithms on Big Data. *In Proceedings of the 2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS)*, 2293–2298.

Vora, S., & Yang, H. (2017). A Comprehensive Study of Eleven Feature Selection Algorithms and their Impact on Text Classification. *Proceeding of Computing Conference 2017*, *July*, 440–449.

Wang, D., Zhang, H., Liu, R., Lv, W., & Wang, D. (2014). t-Test feature selection approach based on term frequency for text. *PATTERN RECOGNITION LETTERS*, *45*(2014), 1–10. https://doi.org/10.1016/j.patrec.2014.02.013

Wang, D., Zhang, H., Lui, R., & Lv, W. (2013). Feature Selection Based on Term Frequency and T-Test for Text Categorization. *Pattern Recognition Letters*, *45*(1), 1–6. https://doi.org/10.1016/j.patrec.2014.02.013

Zhou, H., Han, S., & Liu, Y. (2018). A Novel Feature Selection Approach Based on Document Frequency of Segmented Term Frequency. *IEEE Access*, *6*, 53811–53821. https://doi.org/10.1109/ACCESS.2018.2871109

Zou, H., & Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society. Series B: Statistical Methodology*, *67*(2), 301–320. https://doi.org/10.1111/j.1467-9868.2005.00527.x

Biographies

Abubakar Ado is currently a PhD student in the Faculty of Computer Science and Information Technology at the University Tun Hussein Onn Malaysia (UTHM), and also a lecturer in the Department of Computer Science at YUMSUK, Kano, Nigeria, earned B.Tech. from ATBU, Bauchi, Nigeria, and M. Eng. from LUT, Jinzhou, China. He has published journals and conference papers. Mr Abubakar is a member of NCS, and CPN. His research interests include Machine Learning, Big Data Management, and Soft Computing.

Noor Azah Samsudin is an Associate Professor in Faculty of Computer Science and IT, Universiti Tun Hussein Onn Malaysia (UTHM). She earned her B.Sc. from University of Missouri, Columbia, USA, M.Sc. from National University of Malaysia and Ph.D. from The University of Queensland, Australia. She has published many journals and conference papers on text classification. Her research interests include Machine Learning, Classification, Feature Selection, and ICT innovation in education.

Mustafa Mat Deris is a professor of computer science in the Faculty of Computer Science and Information Technology, UTHM, Malaysia, earned his B.Sc. from UPM, M.Sc. from the University of Bradford, England, and Ph.D. from UPM. He has published more than 270 journals and conference papers. His research interests include distributed databases, data grid, database performance issues, and data mining.

Abdulkadir Abubakar Bichi is a currently a PhD student in the Faculty of Computing at UTM, Malaysia, and also lecturer in the Department of Computer Science at YUMSUK, Kano, Nigeria, earned B.Sc. from KUST, Kano, Nigeria, and M.Sc. From UTM, Malaysia. He has published journals and conference papers. Mr Abdulkadir is a member of NCS, and CPN. His research interests include Natural Language Processing, and Networking.

Aliyu Ahmad is currently a lecturer in the Department of Computer Science at the University Tun Hussein Onn Malaysia (UTHM), earned B.Tech. from ATBU, Bauchi, Nigeria, M.Sc. from LUT, Jinzhou, China, and Ph.D. from UTM, Malaysia. He has published journals and conference papers. His research interests Network Security, and Cloud Computing