




# A high-quality feature selection method based on frequent and correlated items for text classification

Heba Mamdouh Farghaly<sup>1</sup> · Tarek Abd El-Hafeez<sup>1,2</sup> 

Accepted: 12 May 2023 / Published online: 4 June 2023  
© The Author(s) 2023

## Abstract

The feature selection problem is a significant challenge in pattern recognition, especially for classification tasks. The quality of the selected features plays a critical role in building effective models, and poor-quality data can make this process more difficult. This work explores the use of association analysis in data mining to select meaningful features, addressing the issue of duplicated information in the selected features. A novel feature selection technique for text classification is proposed, based on frequent and correlated items. This method considers both relevance and feature interactions, using association as a metric to evaluate the relationship between the target and features. The technique was tested using the SMS spam collecting dataset from the UCI machine learning repository and compared with well-known feature selection methods. The results showed that the proposed technique effectively reduced redundant information while achieving high accuracy (95.155%) using only 6% of the features.

**Keywords** Feature selection · Dimensionality reduction · Text classification · Association rule mining · Feature interaction

## 1 Introduction

The number of complex documents and texts that need a thorough comprehension of data mining techniques to effectively categorize documents in numerous applications has significantly increased recently. Text categorization is a method that may effectively manage and organize texts while also making it easier for users to find essential information fast, making it a significant research topic in the field of information processing.

The process of feature selection is crucial to both text classification and data mining (Peng and Fan 2017; Shang et al. 2013). Text data are unstructured data made up of words. In order to enhance computer processing, textual data are frequently represented using the vector space model (VSM). It is used to transform unstructured data into

structured data and treats the document as a collection of words (features) (Zhang and Duan 2019). Actually, not all feature is useful in creating the document classifier. These features could include those that are unnecessary or irrelevant. Additionally, it negatively impacts the classification outcomes when irrelevant features outnumber relevant ones. In this situation, choosing a subset of the original features frequently enhances classification performance (Sangodiah et al. 2014). As a result, feature selection in the context of text classification is a procedure that seeks to identify a minimal number of significant text features in order to minimize text classification error (Sebastiani 2002).

The current feature selection algorithms have significant difficulties; thus, we urgently require some feature selection techniques that can adapt to large amounts of data and have high accuracy and operational efficiency. Researchers in the field of machine learning are currently paying close attention to feature selection. The following two points summarize the primary reasons: (1) Irrelevant and duplicated features have an impact on the performance of some learning algorithms. According to certain studies, the amount of training data grows exponentially as the number of irrelevant characteristics rises (Langley 1994a; Jain and

---

✉ Tarek Abd El-Hafeez  
tarek@mu.edu.eg

Heba Mamdouh Farghaly  
heba.mamdouh@mu.edu.eg

<sup>1</sup> Department of Computer Science, Faculty of Science, Minia University, EL-Minia, Egypt

<sup>2</sup> Computer Science Unit, Deraya University, EL-Minia, Egypt

Zongker 1997). As a result, feature selection not only decreases computing complexity and increases accuracy of classification, but also aids in the discovery of simpler algorithmic models; (2) issues with high-dimension features in huge data processing are ongoing. The growth of data mining has created a pressing need for large-scale data processing, including gene analysis and information retrieval. High-dimensional feature spaces are inherently unsuitable for machine learning, according to empirical evidence. For some learning algorithms, “dimension disaster” or “combination explosion” is lethal. Therefore, in the case of massive data, feature selection is necessary for dimension reduction.

Feature selection is an important step in many pattern recognition and machine learning tasks, especially in the field of data analysis. The goal of feature selection is to identify the most relevant and informative features in a dataset that can be used to build a model.

One of the main challenges of feature selection is dealing with high-dimensional datasets. With the increasing amount of data being generated, it is becoming more important to find ways to efficiently select the most relevant features to improve the performance of models. In addition, many datasets contain redundant or irrelevant features that can negatively impact the performance of models and increase the risk of overfitting. There are several techniques for feature selection that can be summarized as follows:

- Filter methods: Evaluate each feature individually based on a pre-defined criterion, such as information gain or correlation with the target variable, to select a subset of features.
- Wrapper methods: Evaluate the performance of a machine learning model with a given set of features, and use this information to guide the selection of a new subset of features.
- Embedded methods: Incorporate feature selection as part of the model training process, and use regularization or penalization to select a subset of features.
- Deep learning-based feature selection: In recent years, deep learning techniques have been applied to feature selection to leverage the capacity of neural networks to learn complex representations of data.
- Feature selection for interpretability and fairness: With the increasing importance of transparency and ethical considerations in machine learning, feature selection techniques that prioritize interpretability and fairness have become popular.
- Ensemble feature selection: Ensemble methods have gained popularity in feature selection as they can combine the strengths of multiple feature selection algorithms to produce better results.

- Feature selection in high-dimensional data: As the size of datasets continues to increase, feature selection techniques that can effectively handle high-dimensional data have become increasingly important.
- Feature selection with dimensionality reduction: Combining feature selection with dimensionality reduction techniques such as PCA or t-SNE has become a popular approach to overcome the challenge of high-dimensional data.

The choice of feature selection method depends on the nature of the data, the task being performed, and the resources available. In many cases, a combination of methods may be used to obtain the best results.

In conclusion, feature selection is an essential step in many data analysis tasks and requires careful consideration to identify the most relevant and informative features in a dataset. The selection of the right feature selection technique is critical to building effective models and improving the performance of machine learning algorithms. As a result, we introduce a new feature selection method for text classification in this study in order to decrease the size of the subset of the selected features and increase the classifier’s effectiveness without compromising its accuracy. The suggested approach relies on association analysis using the apriori algorithm (Agrawal and Srikant 1994) to reduce the high-dimensional feature space.

## 2 Research contributions and motivations

The following provides an overview of the study’s main contributions:

Using frequent and Correlated items, the proposed feature selection approach extracts significant features from textual data that

- Can identify the frequent and correlated features that are correlated with each another and highly related to the target variable.
- Select features using an association analysis method.
- Can significantly reduce the number of features that are selected.
- Can eliminate features that are both unnecessary and redundant.

The main advantages of using frequent and correlated items to extracts significant features are as follows:

- Improved model accuracy: By selecting only the most important and relevant features, frequent and correlated items can help improve the accuracy of machine learning models. This is because models trained with fewer features are less likely to overfit the data and can be more robust to noisy or irrelevant features.

- **Reduced computational cost:** Feature selection can also reduce the computational cost of training and deploying machine learning models. This is because models trained with fewer features require less memory and processing power, making it possible to train and deploy them on resource-constrained devices.
- **Improved interpretability:** By selecting only the most important features, frequent and correlated items can make machine learning models more interpretable and easier to understand. This can be especially important in applications where transparency and accountability are important, such as in healthcare or finance.
- **Improved fairness:** By removing irrelevant or biased features, frequent and correlated items can help improve the fairness of machine learning models. This can be especially important in applications where fairness is a concern, such as in criminal justice or hiring.
- **Decreased interpretability:** Models with a large number of features can be difficult to understand and interpret, making it challenging to understand the factors that are driving predictions.
- **Feature redundancy:** With a large number of features, it is possible that some features may be highly correlated with each other, leading to feature redundancy. This can increase the risk of overfitting and can make it challenging to interpret the results of the model.

In order to select significant features, a novel feature selection technique based on frequent and correlated items for text categorization is introduced. The proposed approach integrates association analysis theory with data mining. Interesting relationships between data items can be found via association analysis (Zhao and Liu 2009). By mining frequent and correlated items from the training dataset, the proposed technique aims to find features that are both correlated with each other and strongly related to the target attribute. Also, eliminate features that are unnecessary and redundant in an efficient manner.

### 3 Problem statement

The feature space with high dimensions is one of the most significant concerns in the problems of text categorization. As a result, choosing distinguishing features is crucial for text categorization. There are two main factors for choosing some features over others, according to Forman (2007). Due to scalability, using a smaller subset of features takes less time to compute since employing a large number of features consumes a lot of resources like memory, processing power, storage, network bandwidth, etc. The second factor has to do with how well the algorithm performs. For instance, algorithms work better when features that only add noise rather than additional information are ignored.

Problems associated with using large number of feature sets during the creation of the model can be summarized as follows:

- **Overfitting:** A large number of features can lead to overfitting, where the model becomes too complex and is unable to generalize to new data. This can result in poor performance on unseen data.
- **Computational complexity:** Training and deploying machine learning models with a large number of features can be computationally expensive and may require significant amounts of memory and processing power.
- **Increased risk of bias:** Large feature sets may contain irrelevant or biased features, which can negatively impact the performance and fairness of machine learning models.

### 4 Related work

Since the 1970s, when a significant amount of research was published, feature selection has been an active subject of research. This section reviews a number of recent studies that are related to the feature selection approaches for identifying significant and distinguishing features.

To identify heterogeneous features, Pawening et al. (2016) suggested a method of selecting key features based on mutual information (MI). The suggested method employed a joint mutual information maximization (JMIM) approach that considers the class label while selecting features. Further, to transform non-numerical attributes into numerical ones, it also employed the unsupervised feature transformation (UFT) technique.

An enhanced approach for the chi-square (Chi2) test that incorporates interclass concentration and frequency was presented by the authors (Sun et al. 2017). Three factors, in-class dispersion, frequency, and interclass concentration, were used to improve the Chi2 test.

The authors of Kaoungku et al. (2017) introduced an effective method for data classification combined with feature selection based on association rule mining in order to select features having a substantial impact on the target attribute.

In Qu et al. (2019), association rule-based feature selection (ARFS) was proposed. The frequent 2-item set of the target and feature variables was extracted from the dataset using association rules. The sequential forward selection strategy was coupled to search for feature subsets,

and the performance of the decision trees algorithm was then used as the evaluation criterion for the selected feature subsets.

The authors of Larasati et al. (2019) merged the support vector machine (SVM) classifier with feature weighting and feature selection techniques. By taking a total of  $K = 500$  of the top-ranked variables, the Chi2 was utilized to considerably reduce the number of variables. The weight of each variable that was chosen was determined using the feature weighting method.

In Zhou et al. (2020), a feature selection technique called WCFR (Weight Composition of Feature Relevancy) was proposed. The proposed algorithm utilizes standard deviation to assign weights to the relationships between features and feature sets.

The authors of Zhou et al. (2022) present a feature selection approach based on the combination of mutual information and correlation coefficient (CCMI) to evaluate the connections between various features.

The authors of Wang and Zhou (2021) present a combined feature selection method utilizing the chi-square test and the minimum redundancy approach. Through the chi-square test, the features most closely associated with the classes are selected, and then, a subset of these features with low redundancy is further chosen.

In Pathan et al. (2022), the authors utilize a filter-based feature selection method to identify the most pertinent medical features for predicting heart disease. Additionally, the correlation and interdependence among the various features were investigated.

The experiments mentioned above show that the majority of feature selection methods may effectively detect irrelevant features by utilizing various evaluation functions. However, it mostly focused on either removing the redundant features or taking the interaction of the features into account. In contrast, our approach uses association analysis as a technique for feature selection that seeks to remove the redundant and unnecessary features while also taking into account the interactions between the features. Association rule mining is a well-known technique in data mining. It is the process of deducing correlations between events or items, and these relationships can be expressed as rules for ease of comprehension and convenience when using the rules to forecast the presence of an event or item in the future.

## 5 Methodology

Since poor-quality data can decrease the effectiveness of model creation, data quality is crucial for categorization. Numerous irrelevant features are considered during the model-building process, which is the cause of the problem

with the low performance. In this paper, we thus suggested a new feature subset selection method that looks for important features and also considers feature interaction. In addition, the proposed method use association as a metric rather than more conventional metrics such distance (Kononenko 1994; Liu and Zhang 2016; Anggraeny et al. 2018), dependency (Barraza et al. 2019; Sinayobye et al. 2019), and consistency (Zhao and Liu 2009; Dash and Liu 1997) to assess the relevance between the target attribute and feature(s). The proposed method for classifying texts goes through five stages: text preprocessing, feature extraction, feature selection using the proposed methodology, classification, and finally performance evaluation. Figures 1 and 2 depict the essential steps of the algorithm and the overall structure of the proposed system.

### 5.1 Text preprocessing

In text mining, preprocessing is a vital step and crucial activity that is used to minimize the number of features in a dataset and enhance the performance of the classification strategy in terms of classification accuracy and resource needs. Therefore, most text collections and documents contain irrelevant terms, including stop words, misspellings, and Lang, which need to be removed. Unnecessary features and noise can negatively affect system performance in many algorithms, particularly probabilistic and statistical learning algorithms. For text categorization, we took into account three standard preprocessing techniques: tokenization (Verma et al. 2014), stop-word removal (Saif et al. 2014), and lemmatization (Samir and Lahbib 2018).

### 5.2 Feature extraction

In information retrieval and text mining, feature extraction is a crucial step. It transforms the text from an unstructured original into structured information that a computer can recognize and process by quantifying the distinctive words retrieved from the text that indicate the meaning of the text. Liu et al. (2018). The most used statistics for feature weighting, the TF-IDF approach, is used in this study to select and weight unique words features from the dataset (Soucy and Mineau 2005). The following equation may be used to determine the weight of any word in any document:

$$W_{ji} = tf_{ji} * \log \frac{M}{df_j} \quad (1)$$

where  $W_{ji}$  is denotes the importance of the word  $j$  in the document  $i$ ,  $M$  denotes the number of documents,  $tf_{ji}$  is the frequency of the word  $j$  in document  $i$ , and  $df_j$  denotes the number of documents includes the word  $j$  (Ahuja et al. 2019).

**Fig. 1** Main steps of the proposed text categorization system

### Proposed system algorithm

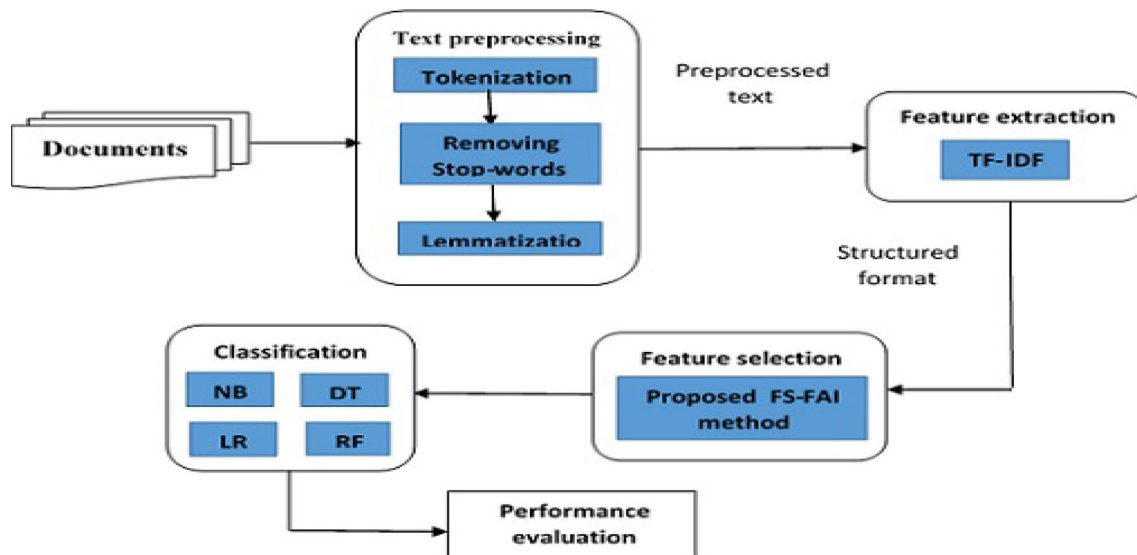
**Input:** D: collection of n documents.

**Output:** classification accuracy rate.

**Begin:**

1. Load dataset D
2. // **Step 1: preprocessing**
3. For each document  $d_i$  in D where  $i = 1:n$
4.     Tokenize document  $d_i$
5.     For each token  $k_j$  in  $d_i$  where  $j = 1:m$
6.         If  $k_j$  is in the list of stop-words: // remove stop words
7.             Del ( $k_j$ )
8.         End if
9.     Lemmatization of token  $k_j$  // apply lemmatization process
10.    End for
11. End for
12. Split D to training set T and test set S.
13. // **Step 2: apply feature extraction for training and testing data**
14. For each term j in document i where  $i=1:n$
15.     Compute TF-IDF weight
16.      $w_{ji} = tf_{ji} * \log(n/df_j)$      // w is the weight matrix
17. End for
18. // **Step 3: apply feature selection on the training dataset**
19.  $F_s = []$      // The selected feature set
20.  $F_s \leftarrow FS\text{-}FAI(T, Min\_supp, \partial)$
21. Fit features to T and S sets
22. // **Step 4: apply classification techniques**
23. Train classifier using T.
24. Make a prediction on S.
25. // **Step 5: evaluate classification performance**
26. Measure the classifiers' performance.

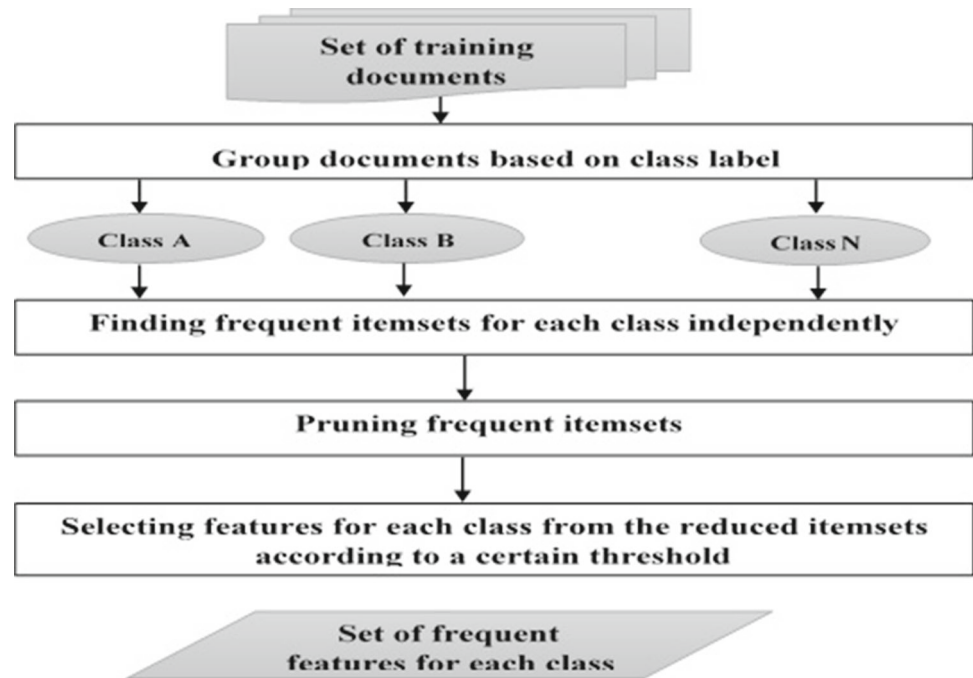
**End**



**Fig. 2** The framework of the proposed text categorization system



**Fig. 3** The proposed method for feature selection



### 5.3 Proposed feature selection method

A new feature selection method is proposed in this section. The proposed method was developed especially for text classification problems. It looks for the important features while taking into account how the features interact with the class feature and with each another. It comprises three steps as shown in Figs. 3 and 4.

1. Identifying frequent items for each class label
2. Pruning frequent items
3. Identifying feature subset.

#### 1. Identifying frequent items for each class label

In this stage, the frequently used items from the pre-processed training dataset are mined. The goal of mining frequently occurring items is to find the undiscovered, interesting relationships between patterns. The frequent set of items are those whose occurrence frequency exceeds a predetermined minimum support (Min\_supp) criteria. The minimum support of the item sets is a parameter that is used to determine the minimum frequency or occurrence of an item set in a dataset. the minimum support of an item set can be calculated using the following equation:

$$\text{Min\_supp}(X) = \frac{(\text{number of transactions in dataset containing } X)}{(\text{total number of transactions in the dataset})} \quad (2)$$

where  $X$  is an item set, and  $\text{Min\_supp}(X)$  is the minimum support of the item set. For example, if there are 100 transactions in a dataset and an item set  $\{A, B\}$  appears in 30 of them, then the support of  $\{A, B\}$  is  $30/100 = 0.3$  or 30%.

At this step, the apriori algorithm is being used on a text document. An apriori algorithm initially determines all frequent sets consists of single item (also known as 1-item sets) that meet a particular Min \_supp criteria. You can continue performing this until you can no longer produce new frequent item sets (Agarwal and Srikant 1994).

In this study, we concentrate on frequently occurring features sets with two or more items. The next step is to prune these sets of items based on specific constraints after collecting the frequent items for each class label.

#### 2. Pruning Frequent set of items

The main goals of the pruning procedure in machine learning are as follows:

- Reducing overfitting: Pruning helps to remove irrelevant or redundant features that may be contributing to overfitting in the model.
- Improving interpretability: Pruning can result in a simpler and more interpretable model, which is easier to understand and explain.
- Reducing computation time: Pruning can reduce the size of the model, which can lead to faster training and inference times.

**Fig. 4** The steps of the proposed method for feature selection**FS-FAI algorithm:****Input:**

T: A training set of documents.  
 Min\_supp: Minimum support threshold  
 Min\_allconf: Threshold for pruning frequent item sets  
 $\partial$ : Threshold for selecting frequent features

**Output:** Frequent\_FS: the set of Frequent features for each class.

**Begin:**

```

1. Let Frequent_FS = []
2. Let C=[c1, c2, .....,cn] where C is the list of all classes
3. For each class c in C do
4.   // Collect document from training dataset that belongs to class c
5.   Let Tc= [] // list of all document that belongs to a certain class
6.   For each d in T:
7.     If ci = c :
8.       Tc.append (d)
9.     End if
10.  End for
11.  // find frequent item sets for each class
12.  F_S = Apriori (Tc, Min_supp)
13.  // prune frequent item sets
14.  PFS= prune(F_S, min_allconf)
15.  // Selecting features from the reduced item sets according to a certain threshold
16.  FFS = Select_Best (PFS, $\partial$ )
17.  Frequent_FS.append (c)
18.  Frequent_FS.append (FFS)
19. End for
20. Return (Frequent_FS).
```

**End.**

- Improving generalization performance: By removing irrelevant or noisy features, pruning can improve the ability of the model to generalize to unseen data.
- Reducing the risk of over-fitting: Pruning can help to reduce the risk of overfitting, especially when dealing with high-dimensional data.
- Balancing between accuracy and complexity: Pruning helps to balance between accuracy and complexity by finding the optimal trade-off between model size and performance.
- Reducing the number of features: Pruning helps to reduce the number of features, which can improve the performance of the model and reduce the risk of overfitting.

Reduce the number of sets of items that were created throughout the frequent item mining process with the pruning approach. This is because some sets of items might not be able to distinguish between classes, which could result in inaccurate classification. As a result, we must prune things to remove redundant and irrelevant data. Figures 5 and 6 show how our suggested algorithm prunes items by implementing the techniques described as follows:

- First, find the associated items:

To calculate the level of mutual association in the set of items, all-confidence is used. All-confidence (Klemettinen et al. 1994) of the set of items  $Y = (k_1, \dots, k_i)$ , denoted as all\_conf ( $k$ ), is defined as follows:

$$\text{All - confidence } (Y) = s(Y) / \max(\text{support}(k_1), \dots, \text{support}(k_i)) \quad (3)$$

According to Eq. 2, all-confidence for each frequent item set is calculated, and then, each item set having an all-confidence value less than or equal to the minimum all-confidence (Min\_allconf) threshold is eliminated.

- Secondly, pruning using the item set redundancy approach:

Using the item set redundancy approach, the set of items is also pruned if it contains another item set.

The correlated and frequent items, whose support and all-confidence are higher than the threshold, respectively, are mined at the end of the pruning process. The effectiveness of the item sets for categorization increases with closer associations between the items. Additionally,

**Fig. 5** Pruning frequent items algorithm

---

**Prune method**

---

**Input:**

$F\_S$ : the set of frequent item sets.

$Min\_allconf$ : threshold for pruning frequent item sets

**Output:** the set of pruned item sets.

**Begin:**

```

1. // Find associated item sets
2. For each item set  $s \in F\_S$ 
3.   all-confidence = compute_all_conf( $s$ );
4.   if all-confidence <  $Min\_allconf$ :
5.     delete  $s$  from  $F\_S$ 
6.   End if
7. End for
8. // Prune more specific item sets with low all confidence
9. For ( $j=1, j \leq Len(F\_S), j++$ ) do
10.  For ( $i=j+1, i \leq Len(F\_S), i++$ ) do
11.    If ( $s_j$ ).is subset ( $s_i$ ):
12.      If all-confidence ( $s_j$ ) < all-confidence ( $s_i$ ):
13.        Del ( $s_j$ ) from  $F\_S$ 
14.      End if
15.    End if
16.  End for
17. End for
18. Return (pruned  $F\_S$ )

```

**End**

---

**Fig. 6** Steps of computing all-confidence for each item set

---

**Compute\_all\_conf method**

---

**Input:**

item set  $s = (a_1, \dots, a_i)$  is a set of values of different attributes.

**Output:** all-confidence ( $s$ ).

**Begin:**

```

1. Let support ( $s$ ) is the support of the item set  $s$ 
2.  $Max\_sup = support(a_1)$ 
3. For ( $i=2, i \leq len(s), i++$ )
4.   If support ( $a_i$ ) >  $Max\_sup$ 
5.      $Max\_sup = support(a_i)$ 
6.   End if
7. End for
8. all-confidence ( $s$ ) = support ( $s$ ) /  $Max\_sup$ 
9. return (all-confidence ( $s$ ))

```

**End**

---

unnecessary item sets are removed in accordance with the idea that an item set should contain other item sets.

### 3. Identifying the Subset of Features

Features are chosen at this phase from the collection of frequent and correlated item sets. As a result, we ought to keep a set of items that are efficient in predicting class labels. For this reason, features that have a proportion of frequency occurrences in the collection of frequent and correlated item sets below the specified minimum frequency threshold are eliminated. The method then returns

the features that are significant for predicting the class attribute after examining their distribution across the item sets taken from the training dataset. Our method for determining the frequent features subset is shown in Fig. 7.

The final feature subset, which retained frequent and significant features while removing redundant and unnecessary features as well as taking feature interaction into account, is determined at the end of the feature selection step.



**Fig. 7** Steps of selecting a frequent feature subset**Select Best method****Input:**

PFS: reduced item sets result from the pruning step.

 $\delta$ : minimum frequency threshold for selecting frequent features**Output:** FFS: the set of frequent features.**Begin:**

```

1. // find list of distinct items from PFS
2. Let items=[]
3. For each item set s ∈ PFS:
4.   For each item i ∈ s:
5.     If i not in items:
6.       items.append(i)
7.     End if
8.   End for
9. End for
10. //compute frequency for each item in
11. For each item i ∈ items:
12.   item_freq = 0
13.   For each item set s ∈ PFS:
14.     If i is in s:
15.       item_freq ++
16.     End if
17.   End for
18.   Compute Freq_percentagei // Freq_percentagei is the frequency percentage of item i
19.   If Freq_percentagei > δ :
20.     FFS.append(i)
21.   End if
22. End for
23. Return (FFS)

```

**End**

## 5.4 Classification process

To properly classify unseen texts, classification algorithms are utilized. Several well-known classifiers are used in this study to evaluate how the selected features by the proposed approach affect classification accuracy, including Naive Bayes (NB), decision trees (DT), and logistic regression (LR).

### 4. Naïve Bayes (NB)

NB (Langley 1994b; Gopal 2019) is a popular classification method that is especially useful for high-dimensional datasets. Although it is a straightforward approach, NB can sometimes achieve better results than more sophisticated classification techniques. This is because NB estimates the likelihood of each input feature or attribute for a particular outcome, and then employs the Bayes rule to calculate the posterior probability for each class  $c_i$  in the dataset.

$$P(C_i|y) = P(y|C_i)P(C_i)/P(y). \quad (4)$$

where

$$P(y) = \sum_j P(y|C_j)P(C_j). \quad (5)$$

where  $P(C_i)$ : the apriori probability of class  $C_i$ .  $P(y)$ : the probability density function of feature  $y$ .  $P(y|C_i)$ : the likelihood of feature  $y$  given that it belongs to class  $C_i$ .  $P(C_i|y)$ : the posterior probability of class  $C_i$  given the observation of  $y$ .

### 5.5 Decision trees (DT)

DT are hierarchical structures commonly used for prediction and classification purposes, and they are considered effective and popular tools. At the top of the hierarchy is the root node, which represents the attribute that is chosen. From there, a branch is created for each available attribute value, and the structure is constructed through a series of if-else conditions. The leaves of the tree hold the result, which in the case of text classification, is a class label (Sohrabi and Karimi 2018).

To determine the homogeneity of a sample, the DT algorithm employs entropy (Zhang et al. 2019). The entropy is calculated using the following mathematical formula:

$$\text{Entropy} = - \sum p \log p \quad (6)$$

where  $p$  is the frequency of the words on the left and right sides of the word  $w$ .

## 5. Logistic Regression (LR)

The LR (Cessie and Houwelingen 1992) is a linear model that is commonly utilized in classification problems. It assesses the connection between the response (dependent) variable and one or more explanatory (independent) variables for a given dataset to determine the importance and strength of the explanatory variables on the response variable. The LR model typically generates probabilities by employing the logistic function, which is also known as the sigmoid function which is given by Shu et al. (2018):

$$f(y) = \frac{L}{1 + e^{-k(y-y_0)}}. \quad (7)$$

where  $e$  is the natural logarithm base,  $L$  is the curve's maximum value,  $y_0$  is the  $y$ -value of the sigmoid's mid-point, and  $k$  is the logistic growth rate or steepness of the curve.

## 5.6 Performance evaluation

The classification accuracy, precision, recall, F-measure and accuracy under ROC curve (AUC) are well-known assessment measures that may be used to assess the effectiveness of the proposed technique (Sokolova et al. 2006).

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}} \quad (8)$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (9)$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (10)$$

$$\text{F1} = 2 * \frac{(\text{Precision} \times \text{Recall})}{(\text{Precision} + \text{Recall})} \quad (11)$$

## 6 Experimental results and analysis

We have performed experiments to evaluate the effectiveness of the proposed feature selection approach. Three different classification techniques, DT, NB, and LR, as well as two different feature selection techniques, analysis of variance (ANOVA) and Chi2 are evaluated using the proposed algorithm. On a PC running 64-bit Windows 7 at a speed of 3 GHz with 4 GB of main RAM, we carried out our testing. Python programming is used to carry out the experiment.

ANOVA (Analysis of Variance) and chi-squared ( $\chi^2$ ) test are both statistical methods used to compare groups or categories, but they are used in different situations and make different assumptions about the data.

ANOVA has the benefit of enabling for the comparison of means across many classes, which enables the analysis of more complicated datasets. This statistical technique is highly flexible and can handle both continuous and categorical data, whereas Chi2 is a straightforward and intuitive test that involves counting the number of observations in each category. It is a widely accepted and well-established method with a rich history of research and it is easy to interpret the results.

The main difference between ANOVA and Chi2test is that ANOVA can be used to test for interactions between variables and can be extended to include other factors such as repeated measures or random effects, while Chi2 test is mainly used to test for independence between two categorical variables.

In summary, ANOVA is used to test for differences in means of continuous data between multiple groups, while Chi2test is used to test for differences in frequencies or proportions of categorical data between multiple groups.

## 6.1 Preparing data for evaluation

In this study, experiments are conducted on the SMS spam collection dataset from the UCI machine learning repository (<http://archive.ics.uci.edu/ml>) to evaluate the classification system. SMS spam collection dataset is one of the most popular text classification datasets. That contains a set of 5196 unique words and 5572 SMS messages that belong to two different classes (ham, and spam).

Lemmatization was applied to convert words from the word vectors generated in the preprocessing stage, which involved removal of irrelevant characters, symbols, and words. 70% of the texts were utilized for training the classifier, by constructing a frequent subset of features. The performance of the classifier was evaluated using the remaining 30% of the texts. Table 1 displays the description of the SMS spam collection dataset.

## 6.2 Results analysis and evaluation

The evaluation of the proposed methodology is presented in this section. The first part focuses on the generation of the frequent feature subset. The second part utilizes a series of tests to gauge the effectiveness of the selected features.

## 6.3 Frequent feature subset generation

The objective of the experiment is to apply the suggested feature selection technique to determine the most suitable features from a list of dataset features. The experiment involves testing various Min\_supp values while keeping the Min\_allconf threshold and the minimal frequency threshold constant, in an effort to attain the best results for

**Table 1** The description of the SMS spam collection dataset description

Class label	No. of documents	No. of documents in the training set	No. of documents in the testing set
Spam	747	516	231
Ham	4825	3384	1441
Total	5572	3900	1672

the proposed method. The Min\_supp threshold values were set to 0.002, 0.003, 0.004, and 0.0045 to extract frequent item sets from the preprocessed SMS spam collection dataset using the apriori algorithm. The minimal frequency threshold was established at 0.048 to identify frequent and correlated features from the pruned item sets, and the Min\_allconf threshold was fixed at 0.13 to reduce the frequent item sets. The Min\_allconf threshold and Minimum Frequency threshold values were determined as the optimal values through a trial-and-error process.

Table 2 shows the number and percentage of frequent features selected using our proposed feature selection method from the preprocessed SMS spam collection training dataset that contains 543 words/features.

The performance of the proposed feature selection method is compared with the two well-known feature selection methods that are ANOVA and Chi2 in the term of the number of selected features. The percentage of choosing the highest scoring features using ANOVA and Chi2 methods is set at 5%, 6%, 8%, 10%, 20%, 30%, and 40%. The number of features selected using ANOVA and Chi2 methods from the preprocessed SMS spam collection training dataset is listed in Table 3.

The number of features chosen by the three feature selection techniques is compared in Tables 2 and 3. It is clear that all feature selection techniques might considerably reduce the number of features chosen. However, our proposed feature selection approach does more than simply try to locate the frequent and correlated features that are related to each other and the target variable. It may also be used to eliminate redundant and unnecessary features. In

**Table 2** Percentage and number of frequent features using the proposed feature selection method for preprocessed SMS spam collection training dataset

Support	No. of selected features	Percentage of selected features (%)
0.002	42	≈ 8
0.003	39	≈ 7
0.004	35	≈ 6.5
0.0045	33	≈ 6

**Table 3** Percentage and number of selected features using the ANOVA and Chi2 feature selection method for the preprocessed SMS spam collection training dataset

Percentage	No. of selected features
≈ 5%	28
≈ 6%	34
≈ 8%	45
≈ 10%	56
≈ 20%	111
≈ 30%	166
≈ 40%	221

contrast to the ANOVA and Chi2 procedures, the ranking criteria are not considered in the variable selection process. The Chi2 approach is used to determine the independence of two variables, whereas the ANOVA method is used to determine the variance between two groups (i.e. features and the target).

The following part will assess how well the features identified using the two methods performed.

## 6.4 Results evaluation

Several studies were carried out to determine the usefulness of our hypotheses and the quality of the selected features. The experiments were carried out using the DT, NB, and LR well-known classification techniques for prediction in three scenarios: (1) without sing feature selection method, (2) with using the well-known feature selection technique that are ANOVA and Chi2 and (3) with using - the proposed feature selection method. The performance of different classifiers was then evaluated. For each classifier, the default parameters were applied.

According to Table 4, which compares the performance of several classification approaches using the first and second scenarios in terms of F-measure, the accuracy of classification, precision, recall and AUC, the best results are denoted in bold format.

From Table 4, when applying the classification technique on the SMS spam collection dataset using the first scenario, we noticed that DT achieves higher performance 94.34% accuracy.

The purpose of this experiment is to compare the first scenario with the second scenario and assess the efficacy of applying classification algorithms using the ANOVA and Chi2 feature selection approach. The training dataset's features were chosen using various percentages of the highest-scoring features.

In Table 4, we can see that when trying to apply the classification using the second scenario, it is typically not

**Table 4** Comparison of different classifiers when using the first and second scenarios in terms of F-measure, accuracy, precision, recall and AUC

			First scenario	Second scenario											
				Feature selection percentage											
				5%		6%		8%		10%		20%		30%	
				ANOVA	Chi2	ANOVA	Chi2	ANOVA	Chi2	ANOVA	Chi2	ANOVA	Chi2	ANOVA	Chi2
NB	Accuracy	65.41	94.79	94.22	94.92	95.05	95.08	95.49	95.47	95.22	95.05	95.01	93.69	93.78	
	F-measure	59.94	95.13	94.63	95.19	95.32	95.24	95.67	95.61	95.38	95.03	94.99	93.42	93.51	
	Precision	75.22	95.88	95.53	95.75	95.89	95.51	96.03	95.89	95.67	95.01	94.97	93.49	93.58	
	Recall	65.41	94.79	94.22	94.92	95.05	95.08	95.49	95.47	95.22	95.05	95.01	93.69	93.78	
	AUC	62.43	93.13	92.06	92.50	92.93	91.42	92.94	92.42	91.77	88.98	88.92	84.65	84.84	
DT	Accuracy	94.34	94.87	94.70	94.87	95.14	95.30	95.27	95.38	95.05	94.87	94.39	94.52	94.79	
	F-measure	94.44	95.20	95.04	95.16	95.36	95.51	95.47	95.57	95.29	94.96	94.51	94.63	94.87	
	Precision	94.58	95.92	95.80	95.76	95.83	95.96	95.87	95.95	95.78	95.08	94.68	94.79	94.98	
	Recall	94.34	94.87	94.70	94.87	95.14	95.30	95.27	95.38	95.05	94.87	94.39	94.52	94.79	
	AUC	88.84	93.21	92.88	92.60	92.58	93.02	92.58	92.84	92.49	89.86	88.96	89.23	89.56	
LR	Accuracy	93.40	92.21	92.25	92.21	92.38	93.03	93.17	93.29	93.61	94.52	94.61	95.31	95.27	
	F-measure	93.39	93.56	93.54	93.56	93.64	94.0	94.13	94.18	94.44	95.13	95.19	95.73	95.69	
	Precision	93.37	96.46	96.30	96.46	96.37	96.18	96.34	96.26	96.42	96.67	96.69	96.87	96.81	
	Recall	93.40	92.21	92.25	92.21	92.38	93.03	93.17	93.29	93.61	94.52	94.61	95.31	95.27	
	AUC	92.06	94.45	93.84	94.45	94.26	94.28	94.85	94.73	95.43	96.52	96.57	96.98	96.73	

**Table 5** The best results achieved for each classification technique using the second scenario

Feature selection method	No. of selected features	Percentage %	Accuracy	Classifier
Chi2	45	8	95.49	NB
ANOVA	56	10	95.38	DT
ANOVA	166	30	95.31	LR

always successful, especially when using low percentages for selecting features. This might be because when using low percentages, some essential features are filtered out, which has an impact on the effectiveness of the classifiers.

For the SMS spam collection dataset, in most cases when using the features selected by two feature selection methods improve the performance of classifiers. Whereas when using feature selection percentage 20% and 30%, the selected features contribute to improving the performance of all classifiers most often. In the case of LR, the performance is enhanced when Chi2 is used with a percentage of 20%, resulting in 94.61% accuracy, and 30%, resulting in 95.27% accuracy. On the other hand, using ANOVA with a percentage of 20% and 30% improves the accuracy to 94.52% and 95.31%, respectively.

These results also demonstrated that, even though the improvement percentage is modest, using the features chosen by two feature selection methods with a percentage

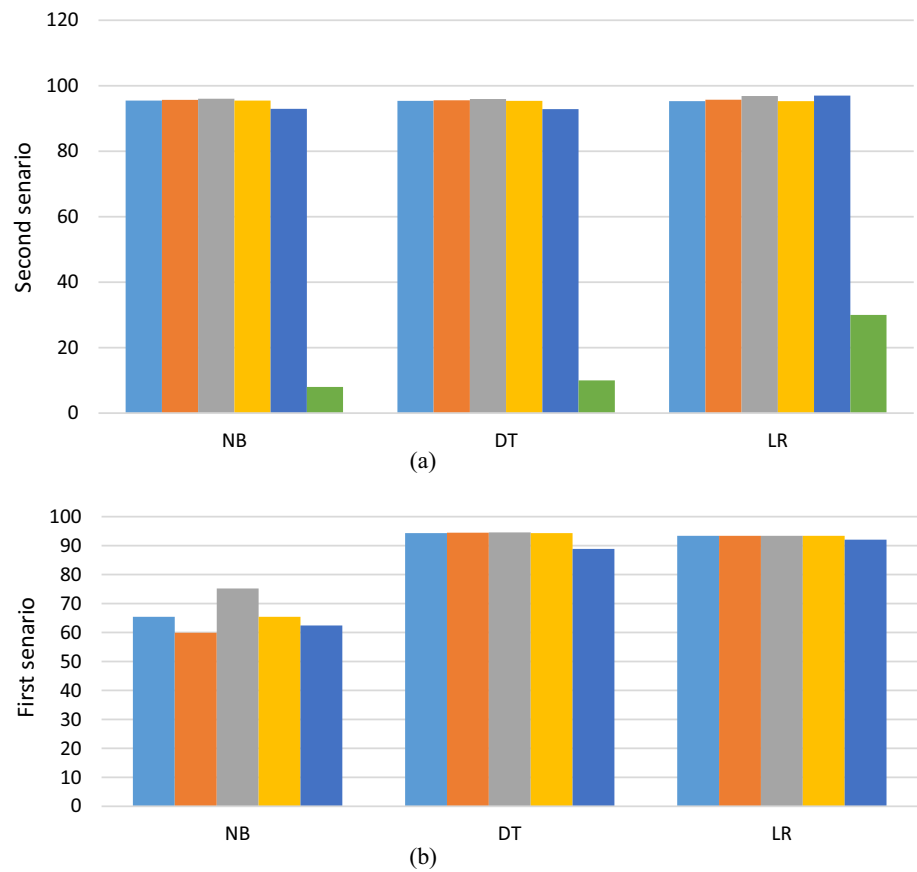
of 20% and 30% can improve the performance of all the classification methods employed. However, the number of features used in this case is significantly lower than that used in the first scenario.

The findings from Table 4 are summarized in Table 5, which also includes the number of features used to train the classifier and the best classification accuracy and F-measure obtained for each classifier using the second scenario.

As shown in Table 5, the results show that the best performance for all classifiers is achieved when using the second scenario with Chi2 and ANOVA feature selection methods.

For the SMS spam collection dataset, the NB obtained the best accuracy of 95.49% using 45 features selected by Chi2 method. The DT obtained the best accuracy of 95.38% using 56 features selected by ANOVA method. The LR obtained the best accuracy of 94.497% using 221 features selected by ANOVA method.

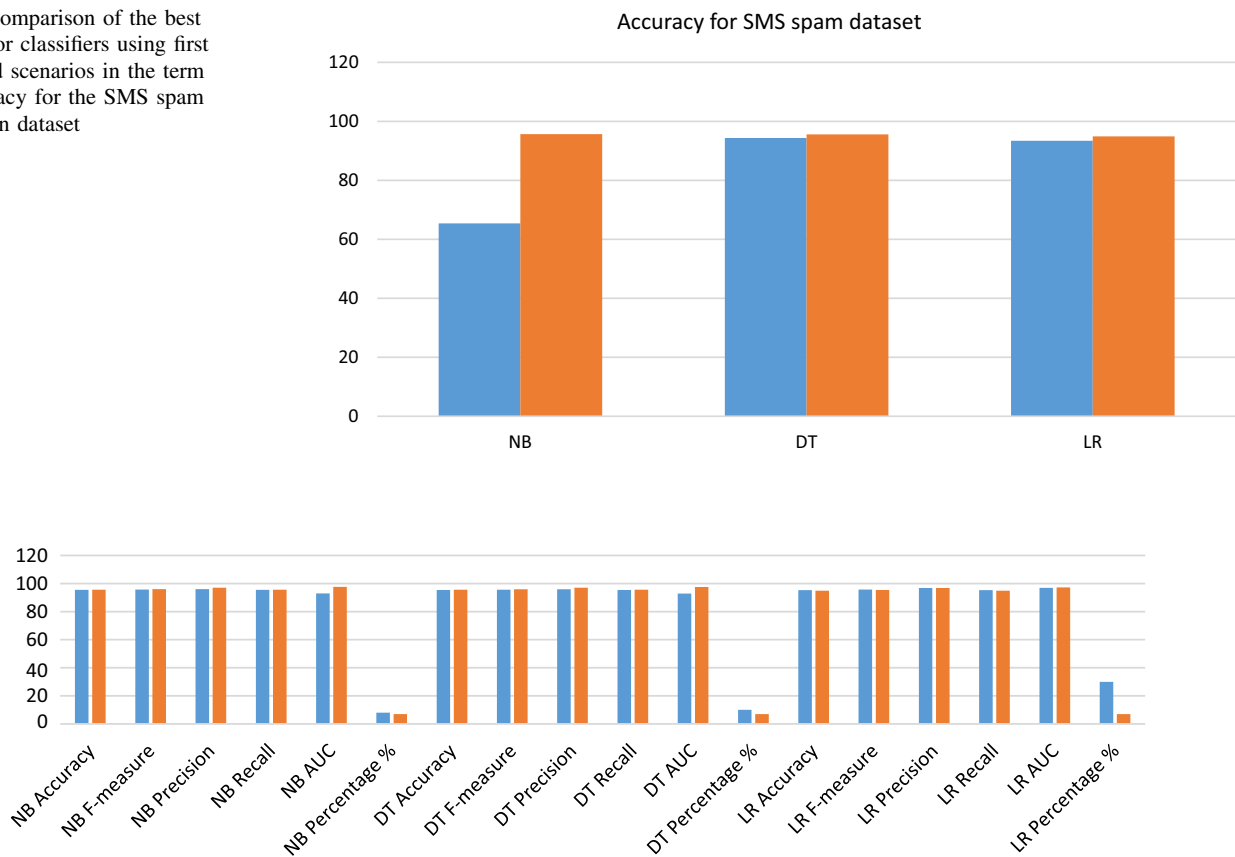
**Fig. 8** Comparative results of classifiers using first and second scenarios using SMS spam collection dataset



**Table 6** Performance comparison of different classifiers using the proposed method with different Min\_supp threshold values for SMS spam collection dataset

	Min_supp threshold			
	0.002	0.003	0.004	0.0045
<i>NB</i>				
Accuracy	95.48	95.64	95.16	95.54
F-measure	95.87	96.003	95.60	95.92
Precision	96.96	97.03	96.85	96.98
Recall	95.48	95.64	95.15	95.54
AUC	97.51	97.59	97.33	97.54
<i>DT</i>				
Accuracy	95.42	95.58	95.1	95.48
F-measure	95.82	95.95	95.56	95.87
Precision	96.94	97.01	96.84	96.96
Recall	95.42	95.57	95.09	95.48
AUC	97.51	97.57	97.31	97.51
<i>LR</i>				
Accuracy	94.62	94.88	94.2	94.49
F-measure	95.19	95.39	94.87	95.09
Precision	96.72	96.79	96.63	96.69
Recall	94.61	94.87	94.19	94.49
AUC	97.06	97.19	96.85	96.99

**Fig. 9** Comparison of the best results for classifiers using first and third scenarios in the term of accuracy for the SMS spam collection dataset



**Fig.10** Comparative results of classifiers using second and third scenarios for SMS spam collection dataset

Figure 8 shows comparative results of classifiers using first and second scenarios. It can be seen that when applying classifiers using the second scenario, the results are improved compared to the first scenario in terms of classification accuracy and F-measure. It is observed that, for the SMS spam collection dataset the NB with Chi2 gives the best performance with 95.49% accuracy using only 8% of features.

The classifiers performance using the third scenario with the proposed feature selection method for SMS spam collection datasets is shown in Table 6.

The purpose of this experiment is to compare the third situation to the first and second scenarios in order to evaluate how well classification techniques applied using the proposed feature selection approach perform. In the proposed feature selection method, features were selected from the SMS spam collection dataset under different values for the Min\_supp that are 0.002, 0.003, 0.004, and 0.0045. For all experiments, the minimum frequency threshold was fixed at 0.048 and the Min\_allconf threshold was fixed at 0.13.

From Table 6, we noticed that when applying classifiers with the proposed approach to select features, their performance is better.

When comparing the outcomes of the third scenario with those of the first scenario, we discovered that the classification approach works significantly better when using the third scenario than when using the algorithms with the first scenario, as shown in Fig. 9. In addition, for the SMS spam collection dataset, we can be observed that the three classifiers using the third scenario have the best performance using only 7% of features.

The second situation (as described in Table 5) and the third scenario given in Table 6 are used to compare the best results obtained when using the classifiers in Fig. 10. When using the third scenario to apply the classifiers, it performs significantly better than when using the second situation to apply the algorithms. But in the case of LR the SMS spam collection dataset, we observed that when applied with ANOVA with 30% of features, it performs better than when applied with the proposed method in terms of classification accuracy. We also noticed that the difference in accuracy between the two methods is not significant, but the percentage of features used in the proposed method is 7%, which is much lower than that used in the ANOVA method.

It can be concluded that for the SMS spam collection dataset, the NB and DT classifiers using the third scenario



have the best performance with accuracy (95.64%) and (95.38%), respectively, using only 7% of features.

## 7 Conclusion

This research introduces a new technique for text classification feature selection that focuses on identifying frequent and correlated items. The aim of the proposed method is to select features that are both frequent and highly correlated with each other and the target attribute. Additionally, it can eliminate unnecessary and redundant features from the feature space.

The results of the experiments conducted in this study are as follows:

- The use of ANOVA and Chi2 procedures sometimes improved the effectiveness of classifiers in terms of accuracy, precision, recall, F-measure and AUC, especially when using fewer features in the second scenario compared to the first scenario.
- When comparing the results of the third scenario with other scenarios using classification algorithms, it was observed that:
- The third scenario achieved the best results in terms of accuracy, precision, recall, F-measure and AUC when using the features obtained from the proposed feature selection approach.
- The proposed approach significantly reduced the number of selected features while still maintaining high quality and efficiency, resulting in lower processing costs and better classification performance.
- For the SMS spam collection datasets, the NB and DT classifiers using the proposed feature selection method outperformed other approaches, achieving an accuracy of (95.64%) and (95.38%), respectively, using only 39 out of 543 features, which is only 7% of the total features.

Finally, this research successfully achieved its objectives and the results demonstrate the efficacy of the proposed feature selection technique for text classification.

**Acknowledgements** Authors sincerely acknowledge Computer Science Department in Faculty of Science, Minia University for the facilities and support.

**Funding** Open access funding provided by The Science, Technology & Innovation Funding Authority (STDF) in cooperation with The Egyptian Knowledge Bank (EKB).

**Availability of data and material** <https://github.com/tarekhemdan/Feature-Selection/blob/main/cs-65924-bbctext.csv>

**Code availability** <https://github.com/tarekhemdan/Feature-Selection>.

## Declarations

**Conflict of interest** The authors declare that they have no conflict of interest.

**Ethical statement** “All procedures performed in studies involving human participants were in accordance with the ethical standards of the institutional and/or national research committee and with the 1964 Helsinki declaration and its later amendments or comparable ethical standards.”

**Consent statement** “Informed consent was obtained from all individual participants included in the study.”

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article’s Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article’s Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## References

- Agarwal R, Srikant R (1994) Fast algorithms for mining association rules. In: Proceedings of the 20th VLDB conference, pp 487–499
- Agrawal R, Srikant R (1994) Fast algorithms for mining association rules. In: Proceedings of the 20th international conference very large data bases, VLDB, vol 1215, pp 487–499
- Ahuja R, Chug A, Kohli S, Gupta S, Ahuja P (2019) The impact of features extraction on the sentiment analysis. *Procedia Comput Sci* 152:341–348
- Anggraeny FT, Purbasari IY, Suryaningsih E (2018) Relief feature selection and Bayesian network model for hepatitis diagnosis. In: Prosiding international conference on information technology and business (ICITB), pp 113–118
- Barraza N, Moro S, Ferreyra M, de la Peña A (2019) Mutual information and sensitivity analysis for feature selection in customer targeting: a comparative study. *J Inf Sci* 45(1):53–67
- Dash M, Liu H (1997) Feature selection for classification. *Intell Data Anal* 1(3):131–156
- Forman G (2007) Feature selection for text classification. *Comput Methods Feature Select* 16:257–274
- Gopal M (2019) Applied machine learning. McGraw-Hill Education, New York
- Jain A, Zongker D (1997) Feature selection: evaluation, application, and small sample performance. *IEEE Trans Pattern Anal Mach Intell* 19(2):153–158
- Kaoungku N, Saksut K, Chanklan R, Kerdprasop K, Kerdprasop N (2017) Data classification based on feature selection with association rule mining. In: Proceedings of the international multicongress of engineers and computer scientists, vol 1
- Klemettinen M, Mannila H, Ronkainen P, Toivonen H, Verkamo AI (1994) Finding interesting rules from large sets of discovered association rules. In: Proceedings of the third international conference on information and knowledge management, pp 401–407

- Kononenko I (1994) Estimating attributes: analysis and extensions of RELIEF. In: European conference on machine learning, pp 171–182
- Langley P (1994a) Selection of relevant features. In: Proceedings of the AAAI fall symposium on relevance, pp 171–182
- Langley P (1994b) Selection of relevant features in machine learning. *Proc AAAI Fall Sympos Relevance* 184:245–271
- Larasati IU, Muslim MA, Arifudin R, Alamsyah A (2019) Improve the accuracy of support vector machine using chi square statistic and term frequency inverse document frequency on movie review sentiment analysis. *Sci J Inf* 6(1):138–149
- Le Cessie S, Van Houwelingen JC (1992) Ridge estimators in logistic regression. *J R Stat Soc Ser C Appl Stat* 41(1):191–201
- Liu Q, Wang J, Zhang D, Yang Y, Wang N (2018) Text features extraction based on TF-IDF associating semantic. In: 2018 IEEE 4th international conference on computer and communications (ICCC), pp 2338–2343
- Liu M, Zhang D (2016) Feature selection with effective distance. *Neurocomputing* 215:100–109
- Pathan MS, Nag A, Pathan MM, Dev S (2022) Analyzing the impact of feature selection on the accuracy of heart disease prediction. *Healthc Anal* 2:100060
- Pawening RE, Darmawan T, Bintana RR, Arifin AZ, Herumurti D (2016) Feature selection methods based on mutual information for classifying heterogeneous features. *Jurnal Ilmu Komputer Dan Informasi* 9(2):106–112
- Peng H, Fan Y (2017) Feature selection by optimizing a lower bound of conditional mutual information. *Inf Sci* 418:652–667
- Qu Y, Fang Y, Yan F (2019) Feature selection algorithm based on association rules. *J Phys Conf Ser* 1168(5):052012
- Saif H, Fernández M, He Y, Alani H (2014) On stop words, filtering and data sparsity for sentiment analysis of twitter, pp 810–817
- Samir A, Lahbib Z (2018) Stemming and lemmatization for information retrieval systems in amazigh language. In: International conference on big data, cloud and applications, pp 222–233
- Sangodiah A, Ahmad R, Ahmad WFW (2014) A review in feature extraction approach in question classification using support vector machine. In: 2014 IEEE international conference on control system, computing and engineering (ICCSCE 2014), pp 536–541
- Sebastiani F (2002) Machine learning in automated text categorization. *ACM Comput Surv CSUR* 34(1):1–47
- Shang C, Li M, Feng S, Jiang Q, Fan J (2013) Feature selection via maximizing global information gain for text classification. *Knowl Based Syst* 54:298–309
- Shu J et al (2018) Clear cell renal cell carcinoma: CT-based radiomics features for the prediction of Fuhrman grade. *Eur J Radiol* 109:8–12
- Sinayobye JO, Kyanda SK, Kiwanuka NF, Musabe R (2019) Hybrid model of correlation based filter feature selection and machine learning classifiers applied on smart meter dataset. In: 2019 IEEE/ACM symposium on software engineering in Africa (SEiA), pp 1–10
- Sohrabi MK, Karimi F (2018) A feature selection approach to detect spam in the facebook social network. *Arab J Sci Eng* 43(2):949–958. <https://doi.org/10.1007/s13369-017-2855-x>
- Sokolova M, Japkowicz N, Szpakowicz S (2006) Beyond accuracy, F-score and ROC: a family of discriminant measures for performance evaluation. In: Australasian joint conference on artificial intelligence, pp 1015–1021
- Soucy P, Mineau GW (2005) Beyond TFIDF weighting for text categorization in the vector space model. *IJCAI* 5:1130–1135
- Sun J, Zhang X, Liao D, Chang V (2017) Efficient method for feature selection in text classification. *Inte Conf Eng Technol ICET* 2017:1–6
- UCI Machine Learning Repository. <http://archive.ics.uci.edu/ml>
- Verma T, Renu R, Gaur D (2014) Tokenization and filtering process in RapidMiner. *Int J Appl Inf Syst* 7(2):16–18
- Wang Y, Zhou C (2021) Feature selection method based on chi-square test and minimum redundancy. In: Emerging trends in intelligent and interactive systems and applications: proceedings of the 5th international conference on intelligent, interactive systems and applications (IISA2020). Springer, pp 171–178
- Zhang L, Duan Q (2019) A feature selection method for multi-label text based on feature importance. *Appl Sci* 9(4):665
- Zhang X, Wang Y, Wu L (2019) Research on cross language text keyword extraction based on information entropy and TextRank. In: 2019 IEEE 3rd information technology, networking, electronic and automation control conference (ITNEC), IEEE, pp 16–19
- Zhao Z, Liu H (2009) Searching for interacting features in subset selection. *Intell Data Anal* 13(2):207–228
- Zhou H, Wang X, Zhang Y (2020) Feature selection based on weighted conditional mutual information. *Appl Comput Inf* (ahead-of-print)
- Zhou H, Wang X, Zhu R (2022) Feature selection based on mutual information with correlation coefficient. *Appl Intell* 52:1–18

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.