International Journal on
Advanced Science
Engineering
Information Technology

Vol.7 (2017) No. 4
ISSN: 2088-5334

# Comparative Analysis of Text Classification Algorithms for Automated Labelling of Quranic Verses

Abdullahi O. Adeleke[#a], Noor A. Samsudin[#b], Aida Mustapha[#c], Nazri M. Nawi[#d]

[#] *Software Engineering Department, Universiti Tun Hussein Onn Malaysia, Parit Raja, Batu Pahat, 86400, Malaysia*
[a] *Corresponding author: hi150007@siswa.uthm.edu.my*
[b] *azah@uthm.edu.my,* [c] *aidam@uthm.edu.my,* [d] *nazri@uthm.edu.my*

*Abstract*— **The ultimate goal of labelling a Quranic verse is to determine its corresponding theme. However, the existing Quranic verse labelling approach is primarily depending on the availability of Quranic scholars who have expertise in Arabic language and Tafseer. In this paper, we propose to automate the labelling task of the Quranic verse using text classification algorithms. We applied three text classification algorithms namely, k-Nearest Neighbour, Support Vector Machine, and Naïve Bayes in automating the labelling procedure. In our experiment with the classification algorithms English translation of the verses are presented as features. The English translation of the verses are then classified as "Shahadah" (the first pillar of Islam) or "Pray" (the second pillar of Islam). It is found that all of the text classification algorithms are capable to achieve more than 70% accuracy in labelling the Quranic verses.**

***Keywords***— **Holy Quran; feature selection techniques; k-Nearest Neighbour; support vector machine; naive bayes**

## I. Introduction

The existence of textual data, be it in offline or online forms, have provided us mass amount of information, which eventually leads to information overload phenomenon [1]. Due to the information overload phenomenon, the study in automated text classification has drawn researchers [2], [3], [4] from many artificial intelligence areas. Text classification is a problem of automatically assigning predefined class membership to unlabelled textual documents [5]. Text classification has been studied to minimize the problem of information overload phenomenon in various problem domains, including news categorization [6], [7], web searching [8], medical document indexing [9], [10], email filtering [11], [12], and sentiment analysis [13], [14].

The information overload phenomenon is also present in understanding content of the Holy Quran [15]. The Holy Quran is the Word of the Almighty God conveying messages to mankind. Although, originally revealed in the Arabic language, due to the massive expansion of Islam for many centuries from the Arabian lands to the entire earth, religious scholars, exegetes, readers of the Quran, and intellectuals have focused on the Quranic study with large scholarly works being produced till date [16]. Translations of the Quran are essentially the interpretations of the scripture of Islam by renowned Muslim scholars such as Ali [17] in English, Hatta [18] in Bahasa Melayu, and Ma Jian [19] in Chinese. However, reading the translations alone may not be easily understood without knowledge sharing from the scholars or experts. The experts are required to be available to justify issues such as the purpose of the verse, the relationship of the verses with the pillars of Islam, and the application of the verse in daily routine. The knowledge is currently recorded in various forms such as labelling the verses some keyword themes prepared manually by the Quran experts.

This study focuses on applying text classification algorithms to automatically label the Quranic verses into two pre-defined categories, which are 'shahadah' and 'prayers'. 'Shahadah' is the Muslim profession of faith ("there is no God but Allah, and Muhammad is the messenger of Allah"). Both 'shahadah' and 'prayers' (more specifically praying five times a day) are the first and second pillars of Islam [20], respectively.

The Quran has about 78,000 words grouped into verses, chapters, quarters, parts etc., by the religious scholars. Within these words (or verses) are messages from the Almighty God. These divine messages could be on issues related to 'iman' (Faith), 'tawheed' (Oneness of Allah), 'seerah' (Stories of previous nations), 'ibadah' (Worship), 'akhlaq' (Virtues) and many other possible classifications for better understanding. Many scholarly works of the Quran have been produced over the years. However, in order to reduce the absolute dependency on individual human experts for determining the related issue, there is a need to automate such issue labelling task using the Quranic translations. In this study, features (or keywords) are analysed from multiple Quran translations in English language by Ali [17], Shakir [21], and Pickthal [17] among others. These features are termed as group-based features as the feature set obtained is a result of analysing the features from the three translations as a whole.

Compare to other textual data, there are very few studies in the classification of Quranic verses based on the English translation [22], [23]. Most researches focus on classification of Quranic verses in Arabic [24], [25], [26], [27].

Hassan et al. [28] implemented a k-Nearest Neighbor (kNN) algorithm to classify the Holy Quran Tafseer verses into predefined categories. To achieve this task, a database of 1000 verses of the Quran was divided into two document sets with the training set consists of 800 verses and the test set consist of 200 verses. Seven predefined categories of the Tafseer texts were chosen (Marriage, Inheritance, Pray, Zakat, Respecting Parents, Halal, and Jihad) after transforming from Arabic to Malay language. The results were evaluated based on Precision and Recall metrics with ‘marriage’ category has the highest recall value of 0.9 and ‘inheritance’ has the lowest recall value of 0.74.

The research work in [25] employed data mining techniques to provide statistical information of analysis on Arabic Quranic verses. The dataset used in the experiment was obtained from Tanzil project website consisting of 114 chapters, 30 parts, 60 groups, 240 hizb-quarter, and 6236 verse. They based their experiments on Term matrices for two selected partitioning methods: Chapters and Parts. From the 114 chapters, they worked on 29 chapters using J48 algorithm. The result for the classiffication of the surahs into Madani, Makki or both showed 73.33% correctly classified instances and 26.66% incorrectly classified instances.

Al-Kabi et al. [29] focused on the automatic classification of the Islam prophet sayings (Hadith) into five predefined categories (classes) namely: Ablution (Wudu’), Fasting, Almsgiving (Zakat), Prayers, and Call to Prayers (Adhaan). Four algorithms (Naive Bayes, Bagging, SVM, LogiBoost) were employed for the classificatiuon task. A dataset of 793 sayings (Hadith) were extracted and used from Sahih Bukhari (one of the two most authentic books of hadith). Classifying hadith is not deterministic due to its structure. Each hadith has two parts (sanad and matn) as well as three sections (quote, action, and report). The dataset was reduced to 474 hadith by filtering out the Quotes section of the sayings manually from the five predefined classes. Accuracy, Recall, Precision, and F-measure evaluation metrics were employed with NB algorithm scored highest with Accuracy rate (56.6038%) and Error rate (43.3962%).

In [26], the work focused on the review of Quranic web portals and their predictions using data mining tools such as Oracle Data Miner (ODM), Weka, SPSS. The dataset was obtained from Alexa’s web information company, a part of Amazon.com company that provides website analytics for all websites counting wise. The specific objective of the research was on studying the access pattern of some websites region wise using classification based data mining tools. The results from the four selected websites (quran.com, islamicity.com, quranexplore.com, tanzil.net) showed the web portal (islamicity.com) had the highest AUC value of 0.69 and Accuracy of 0.87 while the portal (tanzil.net) had the lowest AUC of 0.37 and Accuracy of 0.81.

Akour et al. [24] centered the research work on evaluating similarity of text and documents in Arabic language based on the Holy Quran. Verses of the Quran were used as queries to search for and evaluate similarity. The Measuring Quranic Verses Similarity and Sura Classification (MQVC) approach was employed for retrieving the most similar verses in comparison with a user input verse as a query. The dataset consists of over 2000 verses from the Quran. They applied the MQVC approach randomly on 70 of the 114 chapters of the Holy Quran. The experiment was performed using N-gram technique as well as Machine learning algorithm (LibSVM classifier) for classifying the selected Quran chapters into Makki and Madani chapters. The result recorded the highest precision value of 95% (for chapters 17,32,42) and the lowest precision value of 80% (for chapters 5,13,14,63).

Al-Kabi et al. [30] worked on the classification of different Quranic verses (ayaat) according to their topics using four different classification algorithms (Decision tree, kNN, SVM, NB) with the aim of evaluating their effectiveness. To achieve this, they identified three distinct predefined categories (ignorant of religion; oneness of God; penalty of Apostates). A total number of 1,227 Ayats (verses) were used out of the entire 6,236 Ayats of the Holy Quran to train and evaluate the selected classifiers. Preprocessing phase was performed on the dataset as well as six performance metrics were used to evaluate: True positive (TP), False positive (FP), Precision (P), Recall (R), F-measure (F1), and Receiver Operating Characteristic (ROC). The results show Naive Bayes (NB) scored highest (99.9099% Accuracy) and error rate value (0.0901%) while J48 Decision tree scored lowest (99.5946% Accuracy) and error rate (0.4054%).

Another major challenge in the Quranic verse classification is the high dimensionality of the feature space. Similar to other source documents which are in English, the feature space is comprised of unique words or phrases that occur in the English translated documents, which can be in the count of tens or hundreds of terms. This is prohibitively high for training a classifier. It is then highly desirable to reduce the number of features without sacrificing the classification accuracy of the verses. To do this, there are two possible approach [31]; the ranking features approach and the subset selection approach.

The remainder of this paper is organized as follows: Section II presents the methodology with the description on the dataset, classification experiment as well as the evaluation metrics, Section III reports the experiment results, and finally Section IV concludes with some directions for future work.

## II. Material and Method

This study proposes an improved feature selection approach (otherwise called group-based feature selection) for automatic labelling of Quranic verses. Feature selection is a process commonly used in machine learning to select subset of features available in a data for application of a learning algorithm [32]. Feature selection (FS) is essentially a task of removing irrelevant and/or redundant features from a dataset [31]. In [31], there are two ways used in selecting the best features from a feature space. In this paper, the feature ranking approach is chosen and will be further explained in subsection C. The data used in this study is obtained from the Rasm Uthmani Quran of Hatta [18] as explained in subsection A. Figure 1 illustrates the framework of the proposed GBFS approach employed in this study.

The results obtained from these feature selection algorithms in the classification task are compared in order to evaluate their significant influence on the classification model.

### A. Dataset

The dataset used in this preliminary study is comprised of 200 verses evenly selected from the index ‘Tawheed’ and ‘Solah’ of the Rasm Uthmani Quran of Hatta [18] which corresponded to the class ‘shahadah’ and ‘prayers’ respectively. These verses were randomly split using percentage split into 60% training and 40% testing for the two classes respectively. The dataset comprises of three English language translations of the Quran written by Ali [17], Shakir [21], and Pickthal [17]. Subsection B explains the text preprocessing and feature representation.

### B. Text Preprocessing

In normalizing the text data for the classification task, the term weighting method was employed. To do this, the term frequency ($Tf(t,d)$) measure was used in calculating the frequency of terms (or features) in the dataset. Term frequency is usually denoted with integer 0 to N. In other words, term frequency $Tf(t,d)$ is defined as the number of times a given term t (word/token) appears in a document d [33].

Mathematically, term frequency ($Tf(t,d)$) is shown in equation 1 as:

$$Tf(t,d) = 0.5 + \frac{0.5 \times f(t,d)}{Maximum\ occurences\ of\ words} \quad (1)$$

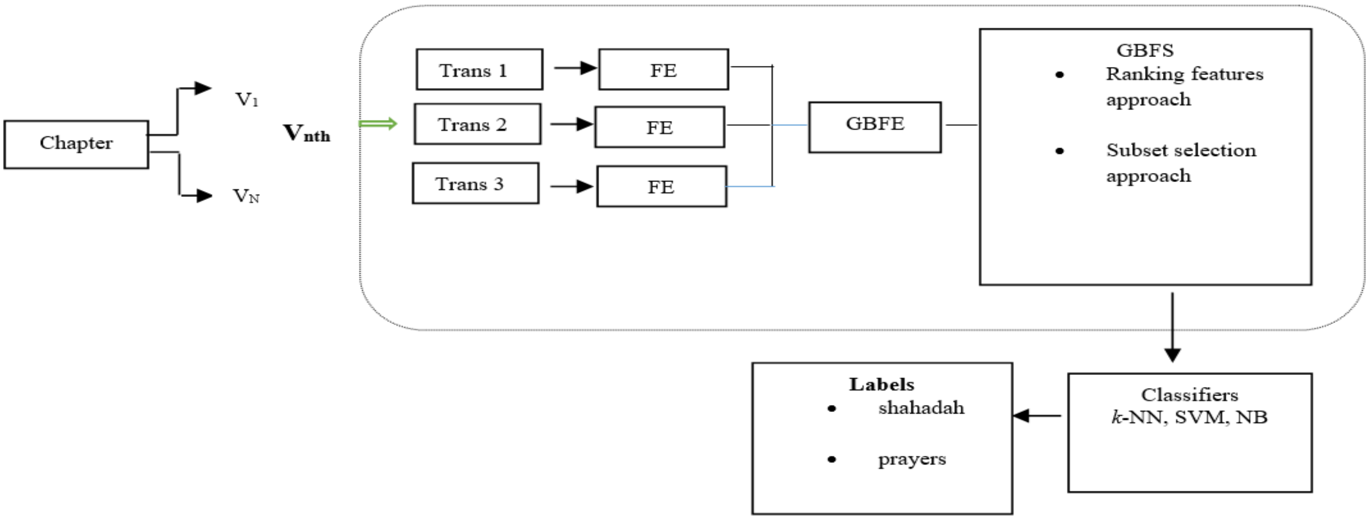

Fig. 2 Proposed Group-Based Feature Selection Approach Framework

The verses in the dataset are represented as vectors comprising of group-based term counts, as opposed to the standard term counts in Information Extraction. To obtain the group-based term counts, the frequency of terms was aggregated based on the natural group within the dataset.

Figure 2 shows the steps involved in obtaining the group-based term counts and an example of the resulting vector that represents a verse. The features (or keywords) of the input verses gotten from Rasm Uthmani Quran of Hatta [18] were automatically extracted by analyzing the frequency of terms in each individual translation before the terms were aggregated for group-based terms/frequency count. This means, the approach was not only looking into presence and absence of terms in a single translation, but instead counted the frequency of the terms used throughout the three translation documents, hence the group-based term count. Eventually, the group-based term count is presented as a feature vector.

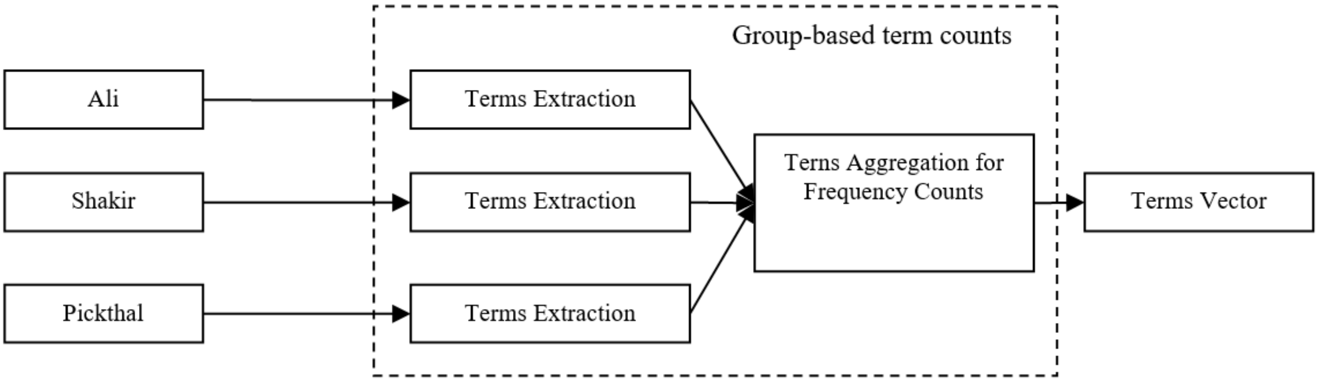

Fig. 2 Group-Based Term Counts for Quranic Text Classification

### C. Feature Selection

Due to high level curse of dimensionality present in data (of most interest text data), which often leads to problems such as overfitting and performance degeneration of the classifiers, the feature selection techniques are employed. As stated in [31], there are two possible ways to feature selection: the ranking features approach and subset selection approach.

The ranking features approach ranks features according to a certain criterion of the feature selection algorithms and the top k features are selected while on the other hand, the subset selection approach selects a minimum subset of features without learning performance deterioration [31]. However, the subset selection approach such as the Wrapper model has a major setback of high computational cost [34]. For this paper, we focused on the ranking features approach (otherwise called Filter model).

The Filter model evaluates features independently of the classification algorithms [35].

This makes the model computationally efficient. Among the most commonly used algorithms in the Filter model family are Pearson correlation coefficients (CFS), Chi-square, and Information Gain (IG) [36]. In this study, the IG and CFS algorithms are implemented with the aim of comparing the effectiveness and impact of the algorithms on the classifiers.

Information Gain (IG) is one of the feature selection methods used in measuring the dependence between features and labels, thereafter then calculates the information gain between the i-th feature $f_i$ and the class labels as shown in equation 2:

$$(f_i, C) - H(f_i) - H(f_i \mid C) \qquad (2)$$

where $H(f_i)$ is the entropy of $f_i$ and $H(f_i \mid C)$ is the entropy of $f_i$ after observing $C$.

Pearson correlation coefficient (also called Linear correlation coefficient) is a method used to evaluate the strength of relationship between two vectors [37]. Given a pair of variables $x$ and $y$, the Pearson correlation coefficient $p$ is given in equation 3 as:

$$p = \frac{\sum_i (x_i - \bar{x}_i)(y_i - \bar{y}_i)}{\sqrt{\sum_i (x_i - \bar{x}_i)^2 \sum_i (y_i - \bar{y}_i)^2}} \qquad (3)$$

where $\bar{x}_i$ and $\bar{y}_i$ are the mean of variables $x$ and $y$ respectively.

The value of $p$ lies between (-1,1) if $x$ and $y$ are linearly dependent (i.e., correlated), and $p$ = 0 if $x$ and $y$ are independent (i.e., uncorrelated) [36]. Thus, to detect if redundancy exist among features, a feature must be strongly correlated to some other features [36].

The feature selection algorithms were implemented on WEKA machine learning tool.

### D. Classification Model

A growing number of data mining techniques have been applied to text classification problem, including the Bayes probabilistic approach [38], [39], decision trees [40], [41], neural networks [42], [43], support vector machines (SVM) [44], [45], [46], and k-nearest neighbor [47], [48]. In this preliminary study, three conventional classification algorithms [49]: nearest neighbor (*k*-NN), SVM, and naïve bayes classifiers are implemented for the labeling task.

The *k*-NN classifier is an instance-based learning algorithm that has shown to be very simple but effective for text classification problem [50]. It is a non-parametric method used in classification and works by calculating the Euclidean distance between points [51]. In classifying a new document $x$, the algorithm ranks the document's neighbors in the training set, and then uses the class of *k* most similar neighbors to predict the class of a new document (also known as majority vote). The Euclidean distance is given in equation 4 as:

$$d(x, x_i) = \sqrt{\sum_{i=1}^{m} (x_j - x_{i,j})^{\wedge}{}_2} \qquad (4)$$

where $x$ is the new point, $x_i$ is the existing point across all input attributes $j$.

The naïve bayes classifier greatly simplify learning by assuming that features are independent given class and has proven effective in many practical applications, including text classification [52]. The classifier is a simple probabilistic model based on the Bayes rule [53]. Given a class $C$, the probabilty of a particular document $d$ to belong to $C$ is given in equation 5 as:

$$P(C_i|\ d) = \frac{P(d\ |\ C_i) * P(C_i)}{P(d)} \qquad (5)$$

SVM is one of the most widely used and applied classification methods. It has been successfully applied to many application domains. SVMs are typically used for learning classification, regression, or ranking function [54]. The algorithm works by searching a seperating hyperplane to seperate between samples with a maximal margin [55]. Equation 6 shows that a hyperplane is:

$$w^T x + b = 0 \qquad (6)$$

To classify an unseen document $d$, the sign of $w^T x + b$ must be known [55]. This is further shown in equation 7 as:

$$w^T x_i + b \geq 1 \text{ or } w^T x_i + b \leq 1 \qquad (7)$$

For each classification algorithm, the percentage split of 60% for the training data and 40% for the test data in training are used in evaluating the classifiers performances respectively. The input to the classifier is a verse represented by a vector of group-based term count. Meanwhile, the output of the classifier is the class; 'shahadah' and 'prayers'.

### *E. Evaluation Metrics*

The objective of this study is to compare the significant influence of the feature selection algorithms on the classification process. The classification experiments were set to measure the accuracy, precision, recall, *f*-measure, area under the receiver operating characteristics curves (AUC), and the ROC curves, across the selected feature selection methods and algorithms on the classifiers. For a better classification result, the AUC value must be closer to 1 [56].

## III. Results and Discussion

In implementing the feature selection algorithms, one of the basic steps is selecting a stopping criterion (also known as threshold) [35]. In other to select the top k ranked features, the threshold must be set. However, it is a very difficult task in selecting a particular threshold for the features selection. Due to this, we experimented ranges of threshold values from 0-1 [49] in selecting the feature subsets as shown in Figure 3 and Figure 4. It could be seen from the figures that the subset selection of features from the features space depends greatly on the threshold and also differ from one feature selection algorithm to another.

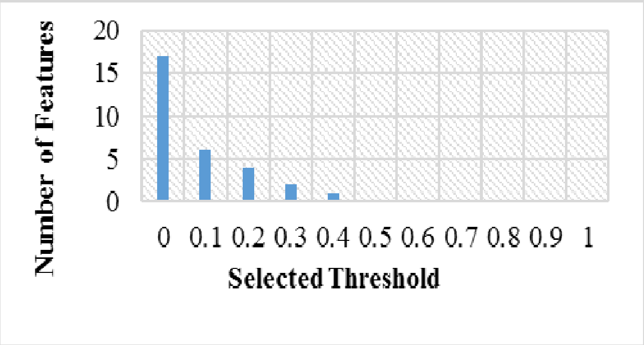

Fig. 3 CFS Feature Selection

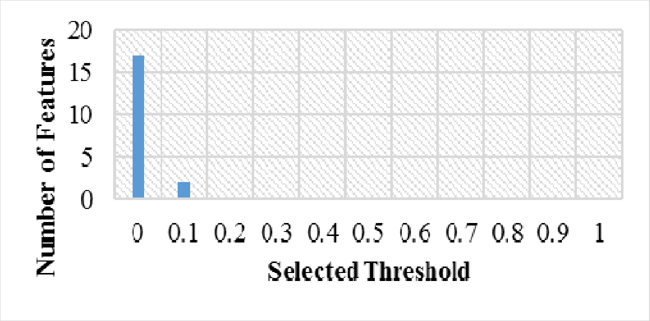

Fig. 4 IG Feature Selection

The CFS algorithm had its threshold set at 0.4 below while the IG algorithm had its threshold from 0.1 downward for significant features subset selection. The reduced subset features are implemented and evaluated on the three conventional classifiers: *k*-NN, NB, and SVM selected for this work as shown in Table 1 to Table 6. The performances of the feature selection algorithms on the classifiers are measured in terms of accuracy, precision, recall, *f*-measure, AUC, and ROC curves.

Table 1 shows the overall classification result when both the CFS and IG feature selection algorithms are implemented. The result obtained in Table 1 shows that at a threshold of 0, the feature selection algorithms (CFS and IG) had impact on the classifiers' performances. The nearest neighbor classifier had the highest AUC value of 0.88 while the SVM had the least AUC value of 0.713. In Table 2 to Table 6, the correlation-based (CFS) and information gain (IG) algorithms had significant impacts on the classifiers. The CFS algorithm had the highest AUC value of 0.863 at a selected threshold of 0.1 using the naïve bayes classifier and the lowest AUC value of 0.7 at a threshold of 0.4 evenly on the three classifiers. The IG feature selection algorithm at the selected threshold of 0.1 had the highest AUC value of 0.779 using the naïve bayes classification algorithm.

From Figure 4, IG Feature selection algorithm stopped at 0.1 threshold. Table 6 shows the classification result based on the number of relevant features selected by the algorithm at 0.1.

TABLE I

GENERAL CLASSIFICATION RESULT (for both CFS and IG FEATURE SELECTION ALGORITHMS)

| **Classifiers** | **Total Number of Features** | **Selected Number of Features** | **Selected Threshold** | **Accuracy** | **Precision** | **Recall** | ***f*-Measure** | **AUC** |
|---|---|---|---|---|---|---|---|---|
| ***k*-NN** | 17 | 17 | 0.0 | 77.5% | 0.786 | 0.775 | 0.773 | 0.880 |
| **SVM** | 17 | 17 | 0.0 | 71.25% | 0.714 | 0.713 | 0.712 | 0.713 |
| **NB** | 17 | 17 | 0.0 | 72.5% | 0.734 | 0.725 | 0.722 | 0.816 |

TABLE II

CLASSIFICATION RESULT (for CFS FEATURE SELECTION ALGORITHM)

| **Classifiers** | **Total Number of Features** | **Selected Number of Features** | **Selected Threshold** | **Accuracy** | **Precision** | **Recall** | ***f*-Measure** | **AUC** |
|---|---|---|---|---|---|---|---|---|
| ***k*-NN** | 17 | 6 | 0.1 | 76.25% | 0.776 | 0.763 | 0.759 | 0.756 |
| **SVM** | 17 | 6 | 0.1 | 77.5% | 0.776 | 0.775 | 0.775 | 0.775 |
| **NB** | 17 | 6 | 0.1 | 71.25% | 0.730 | 0.713 | 0.707 | 0.863 |

TABLE III

CLASSIFICATION RESULT (for CFS FEATURE SELECTION ALGORITHM)

| **Classifiers** | **Total Number of Features** | **Selected Number of Features** | **Selected Threshold** | **Accuracy** | **Precision** | **Recall** | ***f*-Measure** | **AUC** |
|---|---|---|---|---|---|---|---|---|
| ***k*-NN** | 17 | 4 | 0.2 | 76.25% | 0.776 | 0.763 | 0.759 | 0.785 |
| **SVM** | 17 | 4 | 0.2 | 77.5% | 0.776 | 0.775 | 0.775 | 0.775 |
| **NB** | 17 | 4 | 0.2 | 71.25% | 0.730 | 0.713 | 0.707 | 0.854 |

TABLE IV

CLASSIFICATION RESULT (for CFS FEATURE SELECTION ALGORITHM)

| **Classifiers** | **Total Number of Features** | **Selected Number of Features** | **Selected Threshold** | **Accuracy** | **Precision** | **Recall** | ***f*-Measure** | **AUC** |
|---|---|---|---|---|---|---|---|---|
| ***k*-NN** | 17 | 2 | 0.3 | 71.25% | 0.713 | 0.713 | 0.712 | 0.748 |
| **SVM** | 17 | 2 | 0.3 | 73.75% | 0.741 | 0.738 | 0.736 | 0.738 |
| **NB** | 17 | 2 | 0.3 | 70% | 0.813 | 0.700 | 0.670 | 0.779 |

TABLE V

CLASSIFICATION RESULT (for CFS FEATURE SELECTION ALGORITHM)

| **Classifiers** | **Total Number of Features** | **Selected Number of Features** | **Selected Threshold** | **Accuracy** | **Precision** | **Recall** | ***f*-Measure** | **AUC** |
|---|---|---|---|---|---|---|---|---|
| ***k*-NN** | 17 | 1 | 0.4 | 70% | 0.813 | 0.700 | 0.670 | 0.700 |
| **SVM** | 17 | 1 | 0.4 | 70% | 0.813 | 0.700 | 0.670 | 0.700 |
| **NB** | 17 | 1 | 0.4 | 70% | 0.813 | 0.700 | 0.670 | 0.700 |

TABLE VI

CLASSIFICATION RESULT (for IG FEATURE SELECTION ALGORITHM)

| **Classifiers** | **Total Number of Features** | **Selected Number of Features** | **Selected Threshold** | **Accuracy** | **Precision** | **Recall** | ***f*-Measure** | **AUC** |
|---|---|---|---|---|---|---|---|---|
| ***k*-NN** | 17 | 2 | 0.1 | 71.25% | 0.713 | 0.713 | 0.712 | 0.748 |
| **SVM** | 17 | 2 | 0.1 | 73.75% | 0.741 | 0.738 | 0.736 | 0.738 |
| **NB** | 17 | 2 | 0.1 | 70% | 0.813 | 0.700 | 0.670 | 0.779 |

The result obtained in Table 1 shows that at a threshold of 0, the feature selection algorithms (CFS and IG) had impact on the classifiers' performances. The nearest neighbor classifier had the highest AUC value of 0.88 while the SVM had the least AUC value of 0.713. In Table 2 to Table 6, the correlation-based (CFS) and information gain (IG) algorithms had significant impacts on the classifiers. The CFS algorithm had the highest AUC value of 0.863 at a selected threshold of 0.1 using the naïve bayes classifier and the lowest AUC value of 0.7 at a threshold of 0.4 evenly on the three classifiers.

The IG feature selection algorithm at the selected threshold of 0.1 had the highest AUC value of 0.779 using the naïve bayes classification algorithm.

The corresponding receiver operating characteristics (ROC) curves of the classification results taking 'shahadah' as the positive class are shown in Figure 5 to Figure 10.

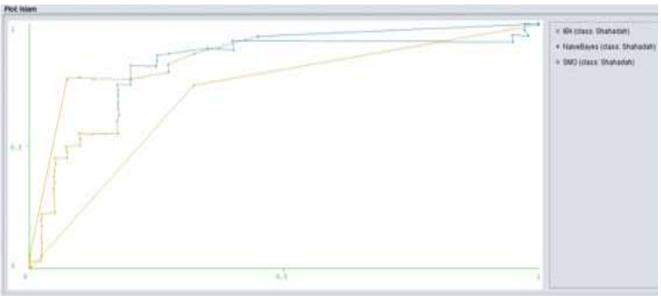

Fig. 5 ROC curve for the general classification result (positive class = 'shahadah')

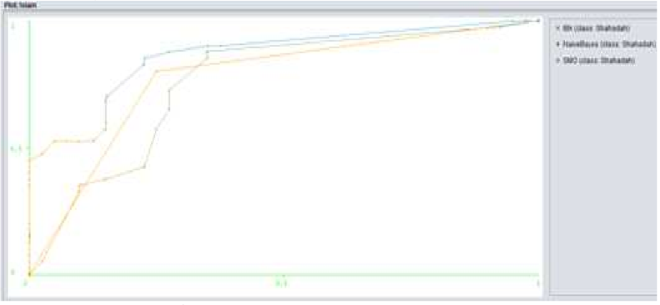

Fig. 6 ROC curve for CFS algorithm at 0.1 threshold (positive class = 'shahadah')

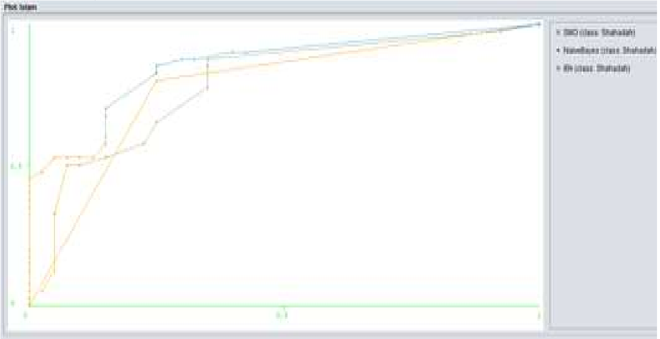

Fig. 7 ROC curve for CFS algorithm at 0.2 threshold (positive class = 'shahadah')

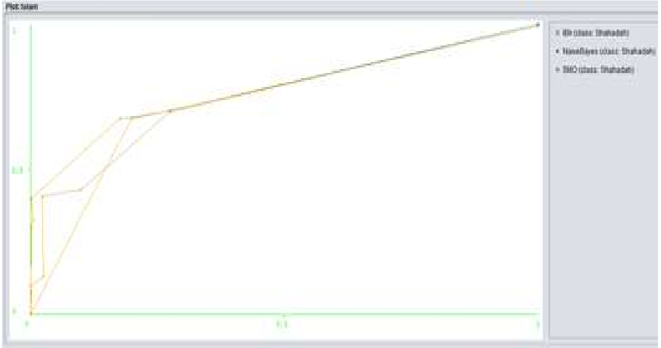

Fig. 8 ROC curve for CFS algorithm at 0.3 threshold (positive class = 'shahadah')

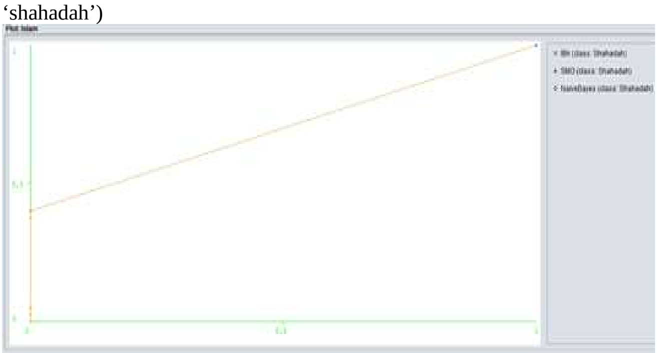

Fig. 9 ROC curve for CFS algorithm at 0.4 threshold (positive class = 'shahadah')

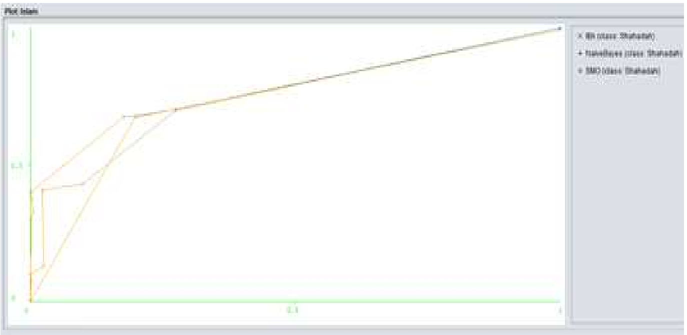

Fig. 10 ROC curve for IG algorithm at 0.1 threshold (positive class = 'shahadah')

## IV. Conclusion

Classifying the Quranic verses into pre-defined categories is an essential task in Quranic studies. In this paper, we presented a feature selection approach to automatically label Quranic verses in order to relate the verses with the five pillars in Islam; the 'Shahada' (profession of faith), the 'Solat' (daily prayers), the 'Zakat' (alms giving), the 'Saum' (fasting during Ramadan), and the 'Hajj' (pilgrimage to Mecca).

The proposed feature selection approach utilized group-based term count to represent the features extracted from three different Quranic translations. The approach adopted two of the common feature selection algorithms to reduce the feature space for 200 randomly selected verses based on the manual index, limited to the first two pillars of Islam.

Finally, the *k*-NN, SVM, and NB classifiers were implemented independently on the Feature selection algorithms to determine class membership for each verse and measure the results in terms of the area under the receiver operating characteristics curve (AUC). The experimental results have shown that the feature selection algorithms employed in the proposed approach had significant impacts on the classifiers implemented in the verse classification task. The lower the selected threshold, the more effective the features subset selection and the better the classifiers' performances. Although, it should be noted that the nature of the dataset determines the number of features to be used.

In conclusion, we hope to extend the proposed GBFS approach in the classification of the entire Holy Quran verses into labels as defined by the Quranic scholars. In addition, this approach could also be implemented on the Prophetic sayings (*Hadith*).

## Acknowledgments

This study was supported in part by a grant from the Ministry of Education of Malaysia, Research Acculturation Grant Scheme (RAGS) Vot R045, a grant from Universiti Tun Hussein Onn Malaysia Vot U611, and in part by a grant from Research Gates IT Solution Sdn. Bhd.

## REFERENCES

[1] T. Herawan, M. Mat Deris, and J. Abawaiy, Eds., *Text Summarization in Android Mobile Devices*, Lecture Notes in Electrical Engineering. Singapore: Springer, 2014, vol. 285.

[2] L. Borrajo, R. Romero, E.L. Iglesias, and C.M.R. Marey, "Improving imbalanced scientific text classification using sampling strategies and dictionaries," *J. Integrative Bioinformatics,* vol. 8, no. 3, pp. 1-15, 2011.

[3] A. Faraz, "An Elaboration of Text Categorization and Automatic Text Classification through Mathematical and Graphical Modelling," *CSEIJ*, vol. 5, no. 2/3, pp. 1-11, 2015.

[4] T.P. Jurka, L. Collingwood, A.E. Boydstun, E. Grossman, and W.V. Atteveldt, "RTextTools: A Supervised Learning Package for Text Classification." *The R Journal*, vol. 5, no. 1, pp. 6-12, 2013.

[5] M.K. Dalal and M.A. Zaveri, "Automatic Text Classification: A Technical Review." *Int. J. of Computer Applications*, vol. 28, no. 2, pp. 37-40, 2011.

[6] M. Hagenau, M. Liebmann, M. Hedwig, and D. Neumann, "Automated News reading: Stock Price Prediction based on Financial News using Context-specific Features," in *45th IEEE (HICSS'12),* 2012, p. 1040.

[7] S.S. Saha, A. Sajjanhar, S. Gao, R. Dew, and Y. Zhao, "Delivering Categorized News items using RSS feeds and Web services," in *10th IEEE (CIT'10),* 2010, p. 698.

[8] M. Radovanovic, M. Ivanovic, and Z. Budimac, "Text Categorization and sorting of Web search results," *Computing and Informatics,* vol. 28, no. 5, pp. 861-893, 2010.

[9] S. Bleik, M. Mishra, J. Huan, and M. Song, "Text Categorization of Biomedical Data Sets using Graph kernels and a controlled vocabulary," *IEEE Transactions on Computational Biology and Bioinformatics*, 2013.

[10] P. Timsina, J. Liu, and O. El-Gayar, "Advanced analytics for the automation of medical systematic reviews." *Information Systems, Frontiers,* vol. 18, no. 2, pp. 237-252, 2016.

[11] T. Gaber, A. Hassanien, N. El-Bendary, and N. Dey, *An E-mail Filtering Approach using Classification Techniques*, in Advanced Intelligent System and Informatics. Switzerland: Springer, 2015, vol. 407.

[12] Q. Yang and F.M. Li, "Support Vector Machine for Customized Email Filtering based on improving Latent Semantic Indexing," in *Proc. ICMLC'05,* 2005, p. 3787.

[13] C.L. Liu, W.H. Hsaio, C.H. Lee, G.C. Lu, and E. Jou, "Movie rating and review summarization in mobile environment." *IEEE Transactions on Systems, Man, and Cybernetics*, 2012.

[14] S. Goyal, "Sentiment analysis of Twitter Data using Text Mining and Hybrid Classification Approach." *Int. J. of Advance Research, Ideas and Innovations in Technology,* vol. 2, no. 5, pp. 1-9, 2016.

[15] S. Nisha, N. Ali, and A.B.M.S. Ali, "Searching Quranic Verses: A keyword based Query solution using .Net Platform," in *IEEE (ICT4M'14),* 2014, p. 1.

[16] J.L. Esposito, *Islam: The Straight Path,* NY: Oxford University Press, 2010.

[17] K. Mohammed, "Assessing English Translations of the Quran." *The Middle East Quarterly,* vol. 12, no. 2, pp. 58-71, 2005.

[18] N. Halo, *Al Fathun Nawa Jilid 1,* Malaysia: Hafizul, 2016.

[19] R. Israeli, Islam in China: Religion, Ethnicity, Culture, and Politics, Maryland: Lexington Books, 2002.

[20] M. Hussain, The Five Pillars of Islam: Laying the Foundations of Divine Love and Service to Humanity, UK: Kube, 2012.

[21] M.H. Shakir, *The Quran with Text and Translation,* NY: Createspace Independent Publishing, 2016.

[22] S.K. Hamed and M.J. Ab Aziz, "A Question Answering System on Holy Quran Translation based on Question Expansion Technique and Neural Network Classification." *J. of Computer Sciences*, vol. 12, no. 3, pp. 169-177, 2016.

[23] B. Hamoud and E. Atwell, "Quran question and answer corpus for data mining with WEKA," in *IEEE (SGCAC'16)*, 2016, p. 211.

[24] M. Akour, I. Alsmadi, and I. Alazzam, "MQVC: Measuring Quranic verses similarity and Sura classification using N-Gram." *WSEAS Transactions on Computers*, vol. 13, pp. 485-491, 2014.

[25] A. Hilal and N. Srinivas, "Analytical of the Initial Holy Quran Letters Based on Data Mining study." *American Int. J. of Research in Formal, Applied & Natural Sciences*, vol. 10, no. 1, pp. 1-8, 2015.

[26] M.K. Siddiqui, S. Naahid, and M.N.I. Khan, "A Review of Quranic Web Portals through Data Mining," *VAWKUM Transactions on Computer Sciences*, vol. 5, no. 2, pp. 1-7, 2014.

[27] M. Alhawarat, "Extracting Topics from the Holy Quran using Generative Models." *Int. J. of Advanced Computer Science and Applications*, vol. 6, no. 12, pp. 288-294, 2015.

[28] G.S. Hassan, S.K. Mohammad, and F.M. Alwan, "Categorization of 'Holy Quran Tafseer' using k-Nearest Neighbour Algorithm." *Int. J. of Computer Applications,* vol. 129, no. 12, pp. 1-6, 2015.

[29] M.N. Al-Kabi, H.A. Wahsheh, I.M. Alsmadi, and A.A. Al-Akhras, "Extended Topical Classification of Hadith Arabic Text." *Int. J. on Islamic Applications in Computer Science and Technology,* vol. 3, no. 3, pp. 13-23, 2015.

[30] M.N. Al-Kabi, B.M. Abu Ata, H.A. Wahsheh, and I.M. Alsamadi, "A Topical Classification of Quranic Arabic Text," in *NOORIC'13*, 2013, p. 272.

[31] H. Liu and H. Motoda, *Computational Methods of Feature Selection,* Boca Raton: CRC Press, 2007.

[32] L. Ladha and T. Deepa, "Feature Selection methods and algorithms." *Int. J. in Computer Science and Engineering,* vol. 3, no. 5, pp. 1787-1797, 2011.

[33] S. Menaka and N. Radha, "Text Classification using Keyword Extraction Technique." *Int. J. of Advanced Research in Computer Science Engineering,* vol. 3, no. 12, pp. 734-740, 2013.

[34] S. Raschka, *Naïve Bayes and Text Classification I,* Cornell University Library, 2015.

[35] C.C. Aggarwal, Ed., *Feature Selection for Classification: A Review*, in Data Classification: Algorithms and Applications. Boca Raton: CRC Press, 2015.

[36] H.F. Eid, A.E. Hassanien, T.H. Kim, and S. Banerjee, "Linear Correlation-Based Feature Selection for Network Intrusion Detection Model," *Advances in Security of Information and Communication Networks,* vol. 381, pp. 240-248, 2013.

[37] J. Xu, B. Tang, H. He, and H. Man, "Semisupervised Feature Selection Based on Relevance and Redundancy Criteria," *IEEE Transactions on Neural Networks and Learning Syst.,* pp. 1-11, 2016.

[38] B. Tang, H. He, P.M. Baggenstoss, and S. Kay, "A Bayesian Classification Approach using Class-Specific Features for Text Categorization." *IEEE Transactions on Knowledge and Data Engineering,* vol. 28, no. 6, pp. 1602-1606, 2016.

[39] A.S. Zharmagambetov and A.A. Pak, "Sentiment analysis of document using deep learning and decision trees," in *Twelve IEEE (ICECCO'15),* 2015, p. 1.

[40] J. Huo, X. Wang, M. Lu, and J. Chen, "Induction of Multi-stage decision tree," in *IEEE (SMC'06),* 2006, p. 835.

[41] J.H. Wang and H.Y. Wang, "Incremental Neural Network Construction for Text Classification," in *IEEE (IS3C'14),* 2014, p. 970.

[42] F. Al Zaghoul and S. Al Dhaheri, "Arabic Text Classification Based on Features Reduction using Artificial Neural Networks," in *15$^{th}$ IEEE (UKSim'13),* 2013, p. 485.

[43] T. Sabbah and A. Selamat, "Support Vector Machine based approach for Quranic words detection in online textual content," in *8th IEEE (MySEC'14),* 2014, p. 325.

[44] H. Sayoud, "Automatic authorship classification of two ancient books: Quran and Hadith," in *IEEE/ACS (AICCSA'14),* 2014, p. 666.

[45] H. Chen and L. Yan, "A Novel Heuristic Text Classification Algorithm based on Support Vector Machine," in *IEEE (CiSE'10),* 2010, p. 1.

[46] M.L. Zhang and Z.H. Zhou, "A k-nearest neighbor based algorithm for multi-label classification," in *IEEE (GRC'05),* 2005, p. 718.

[47] K.R. Townsend, S. Sun, T. Johson, O.G. Attia, P.H. Jones, and J. Zambreno, "k-NN text classification using an FPGA-based sparse matrix vector multiplication accelerator," in *IEEE (EIT'15),* 2015, p. 257.

[48] M. Akhiljabbar, B.L. Deekshatulu, and P. Chandra, "Classification of Heart Disease using k-Nearest Neighbor and Genetic Algorithm," in *Proc. CIMTA'13*, 2013, p. 85.

[49] A.P. James and S. Dimitrijev, "Ranked selection of nearest discriminating features." *Human-centric Computing and Information Sciences.* Springer, 2012.

[50] F.S. Gharehchopogh, S.R. Khaze, and I. Maleki, "A New Approach in Bloggers Classification with Hybrid of k-Nearest Neighbor and Artificial Neural Network Algorithms." *Indian J. of Science and Technology,* vol. 8, no. 3, pp. 237-246, 2015.

[51] L. Dey, S. Chakraborty, A. Biswas, B. Bose, and S. Tiwari, “Sentiment analysis of Review Datasets using Naïve Bayes’ and k-NN Classifiers.” *Int. J. of Information Engineering and Electronic Business,* vol. 4, pp. 54-62, 2016.

[52] H. Yin, K. Tang, Y. Gao, F. Klawonn, M. Leo, X. Yao, Eds., *Fast and accurate sentiment classification using an enhanced Naïve Bayes model*, in Intelligent Data Engineering and Automated Learning. Berlin: Springer, 2013, vol. 8206.

[53] S.S. Nikam, “A Comparative Study of Classification techniques in Data Mining Algorithms.” *Computer Science Technology,* vol. 8, no. 1, pp. 13-19, 2015.

[54] I. Pilaszy, “Text Categorization and Support Vector Machines,” in *Proc. CINTI’05*, 2005, p. 170.

[55] S. Amarappa and S.V. Sathyanarayana, “Data Classification using Support Vector Machine (SVM), a simplified approach.” *Int. J. of Electronics and Computer Science Engineering,* vol. 3, no. 4, pp. 435-445, 2014.

[56] W. Daelemans and K. Morik, Eds., *Improving k-Nearest Neighbor Classification with Distance Functions based on Receiver Operating Characteristics*, in Machine Learning and Knowledge Discovery in Databases, Lecture notes in Artificial Intelligence. Berlin: Springer-Verlag, 2008, vol. 5211.