An Automated Arabic Text Categorization Based on the Frequency Ratio Accumulation

Baraa Sharef [1, 3], Nazlia Omar[1], and Zeyad Sharef [2]
[1]Faculty of Information Science and Technology, University Kebangsaan Malaysia, Malaysia
[2]College of Electronic Engineering, University of Mosul, Iraq
[3]College of Computer Science and Mathematics, University of Mosul, Iraq

Abstract: *Compared to other languages, there is still a limited body of research which has been conducted for the automated Arabic Text Categorization (TC) due to the complex and rich nature of the Arabic language. Most of such research includes supervised Machine Learning (ML) approaches such as Naïve Bayes (NB), K-Nearest Neighbour (KNN), Support Vector Machine and Decision Tree. Most of these techniques have complex mathematical models and do not usually lead to accurate results for Arabic TC. Moreover, all the previous research tended to deal with the Feature Selection (FS) and the classification respectively as independent problems in automatic TC, which led to the cost and complex computational issues. Based on this, the need to apply new techniques suitable for Arabic language and its complex morphology arises. A new approach in the Arabic TC term called the Frequency Ratio Accumulation Method (FRAM), which has a simple mathematical model is applied in this study. The categorization task is combined with a feature processing task. The current research mainly aims at solving the problem of automatic Arabic TC by investigating the FRAM in order to enhance the performance of Arabic TC model. The performance of FRAM classifier is compared with three classifiers based on Bayesian theorem which are called Simple NB, Multi-variant Bernoulli Naïve Bayes (MNB) and Multinomial Naïve Bayes models (MBNB). Based on the findings of the study, the FRAM has outperformed the state of the arts. It's achieved 95.1% macro-F1 value by using unigram word-level representation method.*

Keywords: *Arabic TC, FRAM, automatic TC, text classification.*

Received July 14, 2012; accepted December 6, 2012; published online January 29, 2013

1. Introduction

Today, due to the increasing revolution of technology, and especially, the Internet as the primary source for the last few years of this century, the world is witnessing a huge accumulation of such valuable information which is increasingly growing each day. Although, such a huge accumulation of information is valuable and most of these information are texts, it becomes a problem or a challenge for humans to identify the most relevant information or knowledge. Therefore, Text Categorization (TC) comes to the scene where it plays a crucial role in helping information users overcome such a challenge. As a matter of fact, within the increasing advancement of knowledge and the accumulation of information, many sciences have emerged as to investigate new phenomenon in new areas and for this, TC is concerned with the area of information and knowledge documentation categories. Since information and knowledge stored and divided into categories of documents or texts, the TC assists the users of such information to navigate to the information he/she would like to obtain.

TC, as defined by [17, 23], is the task of automatically assigning selected documents into categories from a pre-defined set of categories. It is also referred to as document classification or topic spotting. It has many applications such as document indexing [4] document organization [16] and hierarchical categorization of web pages [19]. This task is usually solved by combining Information Retrieval (IR) technology and Machine Learning (ML) technology which both work together to assign keywords to the documents and classify them into specific categories [23]. ML helps to categorize the documents automatically and IR represents the text as a feature. The goal of this paper is to categorize electronic Arabic texts to one or more categories automatically and to determine efficiency of the categorization model built.

Generally, there are two problems involved in the processing of automatic TC: The first problem is related to the extraction of feature terms which are recognized as effective keywords in the training phase, and the second problem is concerned with the actual classification of the document using these feature terms in the test phase.

In this paper, a new classification technique in Arabic TC term called the Frequency Ratio Accumulation Method (FRAM) is investigated. It has been proposed by [24] and this method is characterized as classifying the documents without extracting feature

terms in the Feature Selection (FS) stage. Furthermore, this method looks promising to the Arabic TC term. To prove the effectiveness of the proposed method, it is compared with three well-known classifiers based on Bayesian theorem which are called Simple Naïve Bayes (NB), Multi-variant Bernoulli Naïve Bayes (MNB) and Multinomial Naïve Bayes models (MBNB). These classifiers have been applied by [1] on Arabic TC term. Rest of this paper is organized as follows: Section 2 reviews the most related work of Arabic TC. Section 3, describes the methods and materials which used in this study. In section 4, we present the performance measurements which are used to evaluate the categorization models. In section 5, the results and discussion are presented. The paper is concluded in section 5.

2. Related Work

Supervised learning as indicated by [18] is a very popular ML approach, in which, classification patterns derived from a set of labeled examples are learned by TC algorithms, given a huge number of labeled examples (training set), and the task with the aim of building a TC model. Then, the TC model can be used to predict the category of new unseen examples (testing set). Statistical-based algorithms, Bayesian classification, distance-based algorithms, K-Nearest Neighbours (KNN) and decision tree-based methods are some of the different ML algorithms which have been applied for TC [8]. Most of these algorithms applied in different previous studies in TC are designed and tested for documents in English language. However, it is stated that some TC approaches were carried out for TC in other European languages such as German, Italian and Spanish [5], and some others were applied in TC in Chinese and Japanese languages [14, 20]. However, for the core area of the current study, which is TC in Arabic language the work, is still scarce [9, 22]. To our best knowledge there is only one commercial Arabic text categorizer referred as "Sakhr Categorizer" [21].

In comparison to TC conducted in other languages as previously stated, developing TC systems for documents written in Arabic language is a challenging task because of the complex and rich nature of the Arabic language. Arabic language is characterized by its highly inflected and morphologically rich system. Therefore, such complex linguistic system raises serious challenges and obstacles to the task of automatic processing and classification which should be indispensably overcome. Moreover, the use of applied automatic TC techniques for Arabic TC is not an easy task, but it is time and effort consuming. What makes it more complex is that applying some automatic TC techniques for Arabic documents is not as efficient as for English because linguistic structures of the two languages especially in morphology and syntax are totally different. Such reasons seem to be some of the main reasons [11, 22] which can justify the lack of much research in the field of Arabic TC as compared to TC in other languages and especially in English.

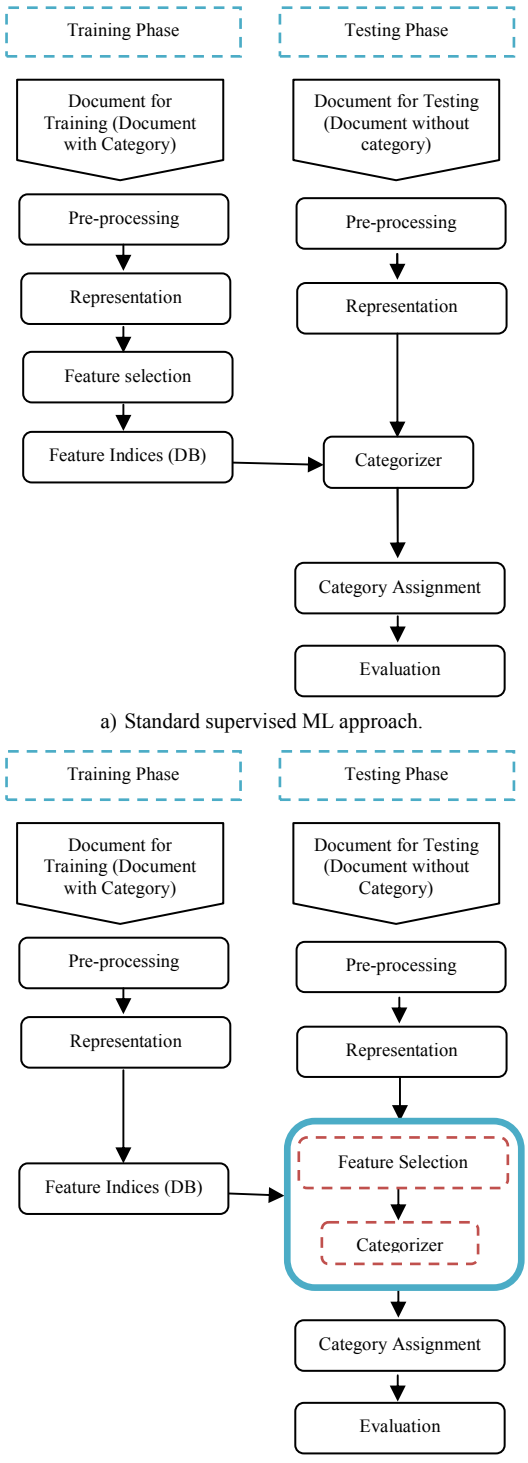
Among the approaches used in the existing research are Simple NB, KNN, Decision Trees and Maximum Entropy. Simple NB classifier is used by [12] for classifying the given Arabic documents into its correct category of the main five categories. Based on the findings obtained from the analysis, it was revealed that the average accuracy over all categories was 68.78%. In [10], maximum entropy method is used to classify the Arabic documents into categories. The classification accuracy obtained in the study is 74.41%. KNN classification technique was investigated in [3] for Arabic TC based on Information Gain (IG) as a FS method. It was found that the best and most accurate result of F-measure obtained by the researcher was 60%. Decision Trees was applied by [13] on Arabic TC. The performance accuracy obtained through Recall, Precision, and F1 measure scored values varied between 0.70 and 0.73 for scientific categories and between 0.37 and 0.43 for literature categories respectively. From the review, we can observe that most of the recent body of the Arabic TC research which is based on ML approaches has not obtained acceptable accuracy in terms of performance.

3. Method and Materials

The methodology that has been adopted to develop a system of Arabic TC is based on supervised ML procedure of automatic TC. Figure 1-a, illustrates the main building blocks of the stages. In the training stage, the documents that are labelled under predefined categories are initially pre-processed in order to eliminate noisy and non-useful terms. Next, features terms that become important keywords are extracted in the training stage from a representation and FS process and getting an indices database, referred to herein as Database (DB), which is used later for the test stage. In the test stage, investigated classifier will be evaluated by classifying a set of pre-categorized documents one by one as uncategorized document, and then measuring the categorization performance by using several standard techniques of performance evaluation.

In this paper we investigated the use of a new classifier called FRAM as proposed by [24] for Arabic TC term. The main aim of this classifier is categorize a new document by calculating the Frequency Ratio (FR) for each feature in the new document from the candidate features in the training phase for each category in the classification phase and then assigning the document under the category that obtain the maximum value of the summation of FR values (more detail in section 3.1). As a result, the FS task will be excluded from the training phase and instead the

features will be assigned to their categories in the classification phase as shown in Figure 1-b.



a) Standard supervised ML approach.

b) Proposed automated Arabic TC system.

Figure 1. Comparison of standard supervised ML approach and proposed automated Arabic TC system.

The problem of automated TC in the present research is classifying a new document d_i into a pre-defined category c_k according to feature terms t_m by using the following notation:

- *Document Set:* $D = \{d_i \mid 1, 2..., I\}$, where d_i is the document and I is total number of all documents.
- *Feature term Set:* $F = \{t_j \mid 1, 2, ... j\}$, where t_j is the word and j is the total number of feature terms contained in all documents. We can express document d_i as a sequence of feature terms in the feature set F. Thus, $d_i = <t_{i1}, t_{i2}, ..., t_{im}>$ where im is the total number of feature terms contained in the each document.
- *Category set:* $C = \{c_k \mid 1, 2, ..., k\}$ where c_k is the category of the total number of categories assigned by k.

3.1. Frequency Ratio Accumulation Categorization Method

FRAM is a new categorization method proposed by [24]. Instead of using FS method for assigning the features generated in the training stage to their appropriate category, FRAM assigns the features that are generated from the new given document to their categories based on the FR of the features that are sorted in the training stage. Assigning the features by using FRAM involves combining it with the classification process. As a result, the computation time for the training stage will be reduced by excluding the FS task. This is unlike the other previous automatic TC approaches which tend to deal with the FS and the categorization successively as independent problems in automatic TC. Moreover, the categorization by this method does not depend on limitation of the number of the training features. The feature terms can be used unlimitedly unlike the other method such as NB classifier which depends on the number of features that affect on the classifier performance.

This method at first calculates the summation of FR of an individual feature term in each category as following:

$$FR(t_n, c_k) = \frac{R(t_n, c_k)}{\sum_{c_k \in C} R(t_n, c_k)} \quad (1)$$

Where, the ratio R of each feature term for each category is calculated by:

$$R(t_n, c_k) = \frac{f_{c_k}(t_n)}{\sum_{t_n \in T} f_{c_k}(t_n)} \quad (2)$$

Here, $f_{ck}(t_n)$ refers to the total frequency of the feature term t_n in a category c_k.

Thus, in the training phase, the FR of all feature terms are calculated and supported in each category. Next, we calculate the category evaluation values or category score, which indicates the possibility that the candidate document in the testing phase belongs to the category as follows:

$$E_{d_i}(c_1) = \sum_{t_n \in d_i} FR(t_n, c_k) \quad (3)$$

Finally, the candidate document d_i is classified into the category $c \wedge k$ for which the category score is the maximum, as follows:

$$c_{\wedge k} = argmax_{c_k \in c} E_{d_i}(c_k) \quad (4)$$

The proposed method i.e. FRAM maintains the FR in the training phase by the total number of the feature terms which are symbolized as *N* and the total number of categories which are symbolized as *K*. Moreover, the category score for each category is calculated by adding the FR when the candidate document in the testing phase includes the feature term and classifies the feature term into the related category for which the evaluation score is the maximum.

The following example explains how FRAM categorizes a new document. Suppose that, the new document is "الصدق كلمة ثمينة" represented by unigram word-level to three features "الصدق" ، "كلمة"، "ثمينة" and the number of occurrences of these features in the training set under each category is shown in Table 1.

Table 1. Number of occurrences of features in the training documents.

Feature	Freq. in C1	Freq. in C2	Freq. in C3	Freq. in C4
الصدق	10	0	50	40
كلمة	65	80	33	20
ثمينة	9	35	70	8

We first calculate the ratios of each feature under each category:

$$R(\text{الصدق},\ c_1) = \frac{f_{c1}(\text{الصدق})}{\sum_{t_n \in T} f_{c1}(t_n)} = \frac{10}{10+65+9} = 0.119$$

$$R(\text{كلمة},\ c_1) = \frac{f_{c1}(\text{كلمة})}{\sum_{t_n \in T} f_{c1}(t_n)} = \frac{65}{10+65+9} = 0.7738$$

$$R(\text{ثمينة},\ c_1) = \frac{f_{c1}(\text{ثمينة})}{\sum_{t_n \in T} f_{c1}(t_n)} = \frac{9}{10+65+9} = 0.107$$

$$R(\text{الصدق},\ c_2) = \frac{f_{c1}(\text{الصدق})}{\sum_{t_n \in T} f_{c2}(t_n)} = \frac{0}{0+80+35} = 0$$

R(كلمة، *c2*)= 0.6956, *R*(ثمينة، *c2*)= 0.6956, *R*(الصدق، *c3*)=0.3268, *R*(كلمة، *c3*)= 0.21569, *R*(ثمينة، c3)= 0.4575, *R*(الصدق، *c4*)= 0.58824, *R*(كلمة، *c4*)= 0.29411, *R*(ثمينة، *c4*)= 0.1176

Then, we calculate the summation of the FR for each feature under each category:

$$E_d(c_1) = \sum_{t_n \in d_i} FR(t_n, c_k)$$
$$= FR(\text{الصدق},\ c_1) + FR(\text{كلمة},\ c_1) + FR(\text{ثمينة},\ c_1)$$
$$= \frac{R(\text{الصدق}, c_1)}{R(\text{الصدق},\ c_1) + R(\text{الصدق}, c_2) + R(\text{الصدق},\ c_3) + R(\text{الصدق},\ c_4)}$$
$$+ \frac{R(\text{كلمة},\ c_1)}{R(\text{كلمة},\ c_1) + R(\text{كلمة},\ c_2) + R(\text{كلمة},\ c_3) + R(\text{كلمة},\ c_4)}$$
$$+ \frac{R(\text{ثمينة}, c_1)}{R(\text{ثمينة}, c_1) + R(\text{ثمينة},\ c_2) + R(\text{ثمينة},\ c_3) + R(\text{ثمينة},\ c_4)} = 0.614674$$

Using the same way to calculate $E_d(c_2)$, $E_d(c_3)$, and $E_d(c_4)$ where:

$$E_d(c_2)=0.6599،\ E_d(c_3)=0.8887،\ and\ E_d(c_4)=0.83669$$

$E_d(c_3)$ is the maximum value, then the document *d* is categorized under the category *C3*.

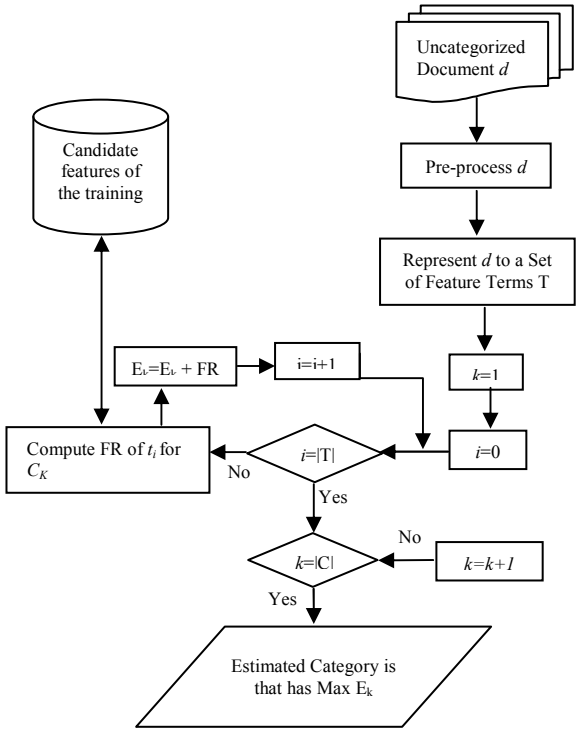


Figure 2. Flow chart of computing the appropriate category for a given document based on FRAM.

The pseudo code of FRAM is shown in Algorithm1.

Algorithm 1. FRAM Pseudo code.

```
Input: A new document d
Preprocess d
Represent d to a set of feature F
For k= 1 to |C|
    E_CK=0
    For i=1 to |F|
        E_CK= E_CK+ FRatio (f_i + c_k)
    Next i
Next k
    Max= E_Cl
For k=1 to |C|
    If E_CK> Max then
        Max= E_Ck
    End if
Next k
Estimated _Category= Max

Function Ratio(f, c)
        T=0
        For i=1 to |F|
            T=T + FRQ(f_i , c)
        Next i
Return FRQ(f, c) / T
Function FRatio(f, c)
        T=0
        For k=1 to |C|
            T=T + Ratio( f , c_k )
        Next i
Return Ratio ( f, c)/T
```

3.2. Text Pre-processing

As indicated by [6], developing a scheme through which the contents of the documents can be scanned

and to which a class is labelled to indicate that the folder best matches the interest of the documents is the main approach for solving problems derived from TC. Thus, the first step known as the text pre-processing is very important as to develop such scheme of any classifier. Moreover, it is pointed out that documents should be pre-processed so that text documents can be categorized by applying ML techniques.

For specifically justifying the importance of this step in TC of the Arabic documents and the morphology complex of the language, the Arabic documents in training and testing phase have been processed as shown in Figure 3 according to the following pre-processing steps:

- *Step 1:* Remove digits and punctuation marks for each text in the Arabic dataset as example Remove (', ', ': ', '? ', ' \ ' …).
- *Step 2:* The non-Arabic words have been filtered.
- *Step 3:* For normalization of some Arabic letters, we have followed [22] and Marwan and Ma shi long (2010) by normalizing the letters "ء" (hamza), "آ" (aleph with madda), "أ" (aleph with hamza on top), "ؤ" (hamza on waw), "إ" (aleph with hamza on the bottom), and "ئ" (hamza on ya) to "ا" (aleph). The reason for this normalization is that all forms of hamza are represented in dictionaries as one form and people often misspell different forms of aleph. We have normalized the letter "ى" to "ي" and the letter "ة" to "ه". The reason behind this normalization is that there is not a single convention for spelling "ى" or "ي"and "ة" or "ه" when they appear at the end of a word. This task is important before carrying out the stemming task especially in the Arabic text because the aim of normalization to reduce the different forms of characters.
- *Step 4:* Remove all the Arabic function words (stop words) which are the words that are not useful in TC systems e.g. pronouns, prepositions and etc., The list of stop word used consists of 368 words as stated by [7].
- *Step 5:* For stemming, we used light Arabic stemming algorithm [7] which is processed as Arabic words to remove all the most common prefixes and suffixes to produce the stem of the Arabic word. The importance of the stemming process is in the categorization and index builders/ searchers because it makes the operations less dependent on particular forms of words and reduces the potential size of vocabularies, which might otherwise have to contain all possible forms.
- *Step 6:* Eliminate all the words with length less than three (such as "بدأ"، "أمر"، "أحد" etc.,).
- *Step 7:* Split the text into tokens consisting only of letters.

The main aim of these processes is to reduce the dimensionality of the Arabic dataset and to transform the documents from plain text to a format which is suitable to the representation process and the other training and categorization tasks.

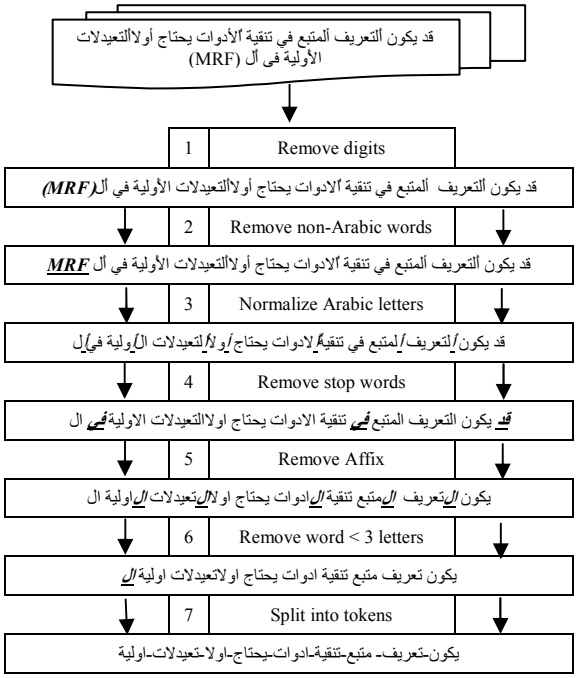


Figure 3. Text pre-processing steps.

3.3. Feature Representation

Since text cannot be directly interpreted by a classifier or by classifier-building algorithm [23], we convert the content of a textual document to feature terms which are composed of character strings which represent a suitable form for learning and categorization to be processed by the computer. These character strings that function well in classification were extracted as feature terms in several previous studies [2, 12]. For this task we applied character-level (N-gram) of 3, 4 and 5 gram and unigram word-level which are commonly used in the previous studies of TC. They are effective as a language-independent method because they do not depend on the meaning of the language and work well in case of noisy text [15]. In this research, a set of N extracted feature terms are expressed as follows:

Feature term set: $T = \{t_n | n = 1, 2, …., N\}$.
N: total number of all features terms.

For unigram word-level, the word is used as a feature term. That is, each feature term t_n corresponds to each single word. On the other hand, using character level N-gram will represent the text to a set of features as sequence words with length n by considering that the single space will be treated as a character. However, instead of using it, the under-score symbol "_" is used. Table 2 shows an example of these methods by representing the Arabic sentence " الصدق كلمة ثمينة".

In the training phase, all possible unigram word-level and N-grams (3, 4 and 5 g) character-level will be generated for each pre-processed document in each

category. Each feature will be weighted by using Term Frequency (TF) method, the number of occurrences of the feature in the document. These features are then sorted according to their frequencies from the most frequent to the least frequent. This will provide the features profile for each category and finally all the most valued features will be saved in the training features DB as the last task of the training phase.

Table 2. Example of text representation.

Representation Type	Generated Features
Character-level 3 gram	الص|الصد|صدق|دق_|ق_ك|_كل|كلم|لمة|مة_|ة_ث|_ثم|ثمي|مين|ينة
Character-level 4 gram	الصد|الصدق|صدق_|دق_ك|ق_كل|_كلم|كلمه|لمة_|مة_ث|ة_ثم|_ثمي|ثمين|مينة
Character-level 5 gram	الصدق|الصدق_|صدق_ك|دق_كل|ق_كلم|_كلمة|كلمة_|لمة_ث|مة_ثم|ة_ثمي|_ثمين|ثمينة
Word-level unigram	الصدق|كلمة|ثمينة

In the testing phase, the given uncategorized document will be pre-processed and represented by using the same pre-processing and representation methods that are used in the training phase to convert the text to a set of features in order to use them in the matching process of the classifier. Usually, feature representation task leads to a huge number of feature terms, may up to ten thousand or hundreds of thousands features. Practically, reducing this highly dimensional task is very difficult due to the fact that each dimension in the feature space is represented by one different and distinguished term or feature appearing in the document collection. This is the major and most challenging difficulty in TC. Therefore, the FS is so important to solve this problem to achieve two main goals. First, it makes the training applied to a categorizer more efficient by decreasing the high dimensionality of effective vocabulary. Second, FS often increases categorization accuracy by reducing rare term. Hence, several variant FS methods as used by [1] with the Bayesian classifiers namely Mutual Information (MI), CHI-Square statistic (CHI), Odds Ratio (OR) and GSS-coefficient (GSS) are compared with our proposed method FRAM where it is combined between the two major processes (classification and FS) as mentioned previously.

4. Performance Measurements

TC performance is always considered in terms of computational efficiency and classification effectiveness. When categorizing a large number of documents into many categories, the computational efficiency of the TC system must be considered. TC effectiveness is measured in terms of Precision, Recall, and the F1 measure. Thus, the effectiveness of our automated Arabic TC system is based on these terms. Each measure is computed by sorting the categorization result into the following:

- *True Positive (TP):* Refers to the number of documents which are correctly assigned to the category.
- *True Negative (TN):* Refers to the number of documents which are not correctly assigned to the category.
- *False Positive (FP):* Refers to the number of documents which are falsely assigned to the category.
- *False Negative (FN):* Refers to the number of documents which are not falsely assigned to the category.

- *Precision:* This measurement use the number of documents which are correctly assigned to a category and it is computed according to the total of positively assigned documents *(TP+FP)* as follows:

$$Presesion(P_i) = \frac{TP_i}{TP_i + FP_i} \quad (5)$$

- *Recall:* Recall use the number of documents which are correctly assigned to a category and it computes according to the total of relevant documents *(TP + TN)* as follows:

$$Recall(R_i) = \frac{TP_i}{TP_i + FP_i} \quad (6)$$

- *F1-Measure:* F1-Measure is combined *Precision* (P_i) and *Recall* (R_i) for category C_i as the following:

$$F1 - measure(F1_i) = \frac{2P_i R_i}{P_i + R_i} = \frac{2TP_i}{FP_i + FN_i + 2TP_i} \quad (7)$$

To evaluate the average performance over all categories, the macro-averaging *F1* have been used by computing arithmetic average *F1* over all categories which defined as:

$$Macro - averagedF1 = \sum_i^{|c|} \frac{F1_i}{|c|} \quad (8)$$

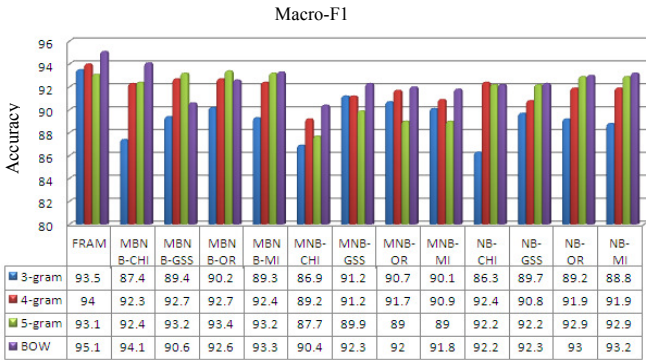
5. Results and Discussion

Since, there is no publicly available Arabic TC corpus to evaluate our experiments, we have used [1] corpus which consists of 3172 documents separated into four categories: Economic, Politics, Arts and Sport. This dataset is divided into training set and test set through randomly selected documents for each category as shown in Table 3. Each document is saved in a separate file within the corresponding category's directory, i.e. this dataset documents are single-labelled.

Table 3 Training and test set for each category.

Categories	No. Documents	Training set	Testing set
Politics	790	430	360
Sport	705	345	360
Art	774	414	360
Economic	903	543	360

Figure 4 depicts Macro-F1 values for all categories. The proposed method FRAM outperforms the state-of-the-arts (MBNB, MNB and NB) with several feature selections methods overall by using 3 gram character-level representation method where FRAM achieved best performance of macro-averaging precision (93.6%) compared to the best performance (91.2%) which is achieved by MNB with GSS FS method when the number of feature is 1000 or 1200 in the same morphological analysis as 3 gram character-level representation method. There is a difference of 2.4% between the performance of the state-of-the-art (MNB with GSS) and that by the proposed method FRAM. This difference may be due to the high frequency that obtained in this type of representation for each feature where the performance of the proposed method depends on the FR of the features in each category.



	FRAM	MBNB-CHI	MBNB-GSS	MBNB-OR	MBNB-MI	MNB-CHI	MNB-GSS	MNB-OR	MNB-MI	NB-CHI	NB-GSS	NB-OR	NB-MI
3-gram	93.5	87.4	89.4	90.2	89.3	86.9	91.2	90.7	90.1	86.3	89.7	89.2	88.8
4-gram	94	92.3	92.7	92.7	92.4	89.2	91.2	91.7	90.9	92.4	90.8	91.9	91.9
5-gram	93.1	92.4	93.2	93.4	93.2	87.7	89.9	89	89	92.2	92.2	92.9	92.9
BOW	95.1	94.1	90.6	92.6	93.3	90.4	92.3	92	91.8	92.2	92.3	93	93.2

Figure 4. Macro-F1.

For the 4 gram character-level, FRAM obtains the best value of Macro-F1 (94%) compared with Bayesian learning classifiers, where MBNB get the best Macro-F1 value (92.7%) by using OR or GSS for selecting the features when the number of features is 1200. A difference of (1.3%) was observed between the performance of state-of-the-art (MBNB with OR or GSS) and that by proposed method FRAM in the same morphological analysis as 4 gram representation process. Thus, we are able to confirm the effect of FRAM to be greater because the result of the high frequency that is obtained in this type of representation for each feature where the performance of the proposed method depends on the FR of the features.

On the other hand, FRAM achieved the worst performance (93.1%) by using 5-gram character-level representation obtained low frequency of the features where the proposed method depend on the frequency representation based on Macro-F1 values among all the used text representation methods because this type of ratio of each features. Nevertheless, it outperforms MNB and NB by using all FS methods expect MBNB classifier method where it achieved the best Macro-F1 value (93.4%) by using 1000 features selected by OR in the same morphological analysis as 5-gram representation.

In respect of using Bag-Of-Word (BOW) representation type, FRAM achieved the best performance with (95.1%) Macro-F1 outperformed MBNB which it obtains the best performance of Macro-F1 equals (94.1%) with the use of 400 features selected by CHI FS method. Using the BOW as feature representation leads to the best performance in each classifier overall in the text representation techniques. This may be due to the simplicity and efficiency of it since only the frequency of a word occurring in a document is recorded, while all the structure and the ordering of the words are ignored. This is unlike character level n-gram which presents the text as dependent features.

The experiments findings prove that using FRAM leads to the best performance of Arabic TC. Three Bayesian learning classifiers; MBNB, MNB and NB, with several FS methods; MI, OR, GSS and CHI, are used for comparing the categorization performance of FRAM. The results proved that FRAM outperforms Bayesian learning classifiers overall. The reason of the exceeding performance of FRAM is because FRAM is estimating the appropriate category based on computing the FR of the feature terms of the given document in the whole features, under that category, as provided in the training phase. Whereas, in Bayesian learning, it is necessary to carry out FS methods to select and reduce the number of FS which could have low frequency.

The limitation of the number of features under the training set may contribute to the reason of the poor performance of Bayesian learning compared to FRAM where it is able to classify the given documents by using unlimited number of training features. For example, if by chance, the important features are not selected in the limited set of features for a certain category but these features are included in the target document. The probability of classifying the target document into that category will be very low.

Furthermore, using character level n-gram as shown in Figure 3 leads to the accepted categorization performance of FRAM because it computes the summation of the features frequencies ratio by considering that the features are related. Unlike Bayesian learning, this is based on the independency of computing the probability of the features given a category and basically using n-gram language will present the text to dependent features. Furthermore, the computation time of learning FRAM classifier will be less than that needed to learn Bayesian classifiers because there is no need to apply FS methods for FRAM while it is very necessary for Bayesian learning.

6. Conclusions

The performance of Arabic TC has been enhanced by applying the FR Accumulation Method FRAM

compared to Bayesian learning classifiers namely Simple NB, MNB and MBNB which are the major methods of supervised ML. The advantage of the proposed method FRAM is that it deals with FS and categorization in one process which leads to reduce the computational operations of Arabic TC system unlike the other methods which deal with FS and classification as a major process of automated TC. Moreover, the categorization based on this method does not depend on the limitations of the number of the training features. The feature terms can be used unlimitedly unlike the other method such as the Bayesian learning classifier which depends on the number of features that affect the classifier performance. As a result, the proposed classifier is suitable for Arabic language and its complex morphology.

For future research, improvement of the performance of the FRAM method on Arabic TC approach by using the other suitable stemmer algorithms for Arabic language can be carried out. Furthermore, investigation of the suitability of other methods of classification and comparison of their performance could be conducted in the future research. This work can be extended by applying the same classifier for other languages.

References

[1] Al-Salemi B. and Aziz M., "Statistical Bayesian Learning for Automatic Arabic Text Categorization," *Journal of Computer Science*, vol. 7, no. 1, pp. 39-45, 2011.

[2] Al-Shalabi R. and Obeidat R., "Improving KNN Arabic Text Classification with N-Grams Based Document Indexing," *in Proceedings of the 6th International Conference on Informatics and Systems*, Cairo, Egypt, pp. 108-112, 2008.

[3] Al-Shalabi R., Kanaan G., and Gharaibeh M., "Arabic Text Categorization Using K-NN Algorithm," *in Proceedings of the 4th International Multi-conference on Computer Science and Information Technology*, Jordan, vol. 4, pp. 1-8, 2006.

[4] Biebricher P., Fuhr N., Lustig G., and Schwantner M., "The Automatic Indexing System AIR/PHYS. from Research to Application," *in Proceedings of the 11th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, France, pp. 333-342, 1988.

[5] Ciravegna F., "Flexible Text Classification for Financial Applications: The FACILE System," *in Proceedings of the 14th European Conference on Artificial Intelligence*, Berlin, Germany, pp. 696-700, 2000.

[6] Croft W., Donald M., and Trevor S., *Search Engines Information Retrieval in Practice*, Pearson Education, 2010.

[7] Darwish K. and Douglas O., "CLIR Experiments at Maryland for TREC-2002: Evidence Combination for Arabic-English Retrieval," *Technical Report*, University of Maryland, USA, 2002.

[8] Dunham M., *Data Mining: Introductory and Advanced Topics*, Prentice Hall, pp. 51-76, 2003.

[9] Duwairi R. and Al-Zubaidi Rania., "A Hierarchical K-NN Classifier for Textual Data," *the International Arab Journal of Information Technology*, vol. 8, no. 3, pp. 251-259, 2011.

[10] El-Halees A., "Mining Arabic Association Rules for Text Classification," *in Proceedings of the 1st International Conference on Mathematical Sciences*, Palestine, pp. 15-17, 2006.

[11] El-Halees A., "Arabic Text Classification Using Maximum Entropy," *the Islamic University Journal*, vol. 15, no. 1, pp. 157-167, 2007.

[12] El-Kourdi M., Bensaid A., and Rachidi T., "Automatic Arabic Document Categorization Based on the Naïve Bayes Algorithm," *in Proceedings of the Workshop on Computational Approaches to Arabic Script-Based Languages*, Stroudsburg, USA, pp. 51-58, 2004.

[13] Harrag F., El-Qawasmeh E., and Pichappan P., "Improving Arabic Text Categorization Using Decision Trees," *in Proceedings of the 1st International Conference on Networked Digital Technologies*, Ostrava, Czech Republic, pp. 110-115. 2009.

[14] He J., Tan A., and Tan C., "On Machine Learning Methods for Chinese Document Categorization," *Applied Intelligence*, vol. 18, no. 3, pp. 311-322, 2003.

[15] Khreisat L., "A Machine Learning Approach for Arabic Text Classification Using N-Gram Frequency Statistics," *the Journal of Informetrics*, vol. 3, no. 1, pp. 72-77, 2006.

[16] Larkey L., "A Patent Search and Classification System," *in Proceedings of the 4th ACM Conference on Digital Libraries*, California, USA, pp. 179-187, 1999.

[17] Manning C. and Schutze H., *Foundations of Statistical Natural Language Processing*, MIT Press, USA, 1999.

[18] Moens M., *Information Extraction, Algorithms and Prospects in the Retrieval Context*, Springer, Berlin, 2006.

[19] Ozgur L., "Adaptive Anti-Spam Filtering Based on Turkish Morphological Analysis Artificial Neural Networks and Bayes Filtering," *Master's Thesis,* Bogazici University, Turkey, 2003.

[20] Peng F., Huang X., Dale S., and Shaojun W., "Text Classification in Asian Languages without Word Segmentation," *in Proceedings of the 6th*

International Workshop on Information Retrieval with Asian Languages, Japan, vol. 11, pp. 41-48, 2003.

[21] Sakhr Software Company's., available at: www.sakhr.com, last visited 2004.

[22] Samir M., Ata W., and Darwish N., "A New Technique for Automatic Text Categorization for Arabic Documents," *in Proceedings of the 5th Conference of the Internet and Information Technology in Modern Organizations*, Egypt, pp. 13-15, 2005.

[23] Sebastiani F., "Machine Learning in Automated Text Categorization," *ACM Computing Surveys*, vol. 34, no. 1, pp. 1-47, 2002.

[24] Suzuki M. and Hirasawa S., "Text Categorization Based on the Ratio of Word Frequency in Each Categories," *in Proceedings of IEEE International Conference on Systems Man and Cybernetics*, Montreal, Canada, pp. 3535-3540, 2007.

Baraa Sharef obtained his MS in information technology from the National University of Malaysia since 2011. His main research interest is in the area of information retrieval, natural language processing, communication and network technology. Currently, he is a PhD candidate on the Department of Computer Science/ Faculty of Information Technology/ National University of Malaysia. He is a reviewer in the (SCIRP and APACE).

Nazlia Omar is currently an associate professor at the School of Computer Science, Faculty of Information Science and Technology, University Kebangsaan Malaysia. She holds her PhD in computer science from the University of Ulster, UK. Her main research interest is in the area of natural language processing, database and information systems.

Zeyad Sharef is currently an assistant lecturer at College of Electronic Engineering, University of Mosul, Iraq. He received his M.Eng. in computer and communication engineering from the National University of Malaysia since 2009. He also leads and teaches modules at BSc levels in computer engineering. The areas of his interest are microcontroller systems, networking and mobile communications.