Contents lists available at ScienceDirect

Journal of King Saud University – Computer and Information Sciences

journal homepage: www.sciencedirect.com

Text representation and classification based on bi-gram alphabet

Fatma Elghannam

Electronics Research Institute, Cairo, Egypt

ARTICLE INFO

Article history:
Received 23 April 2018
Revised 6 January 2019
Accepted 12 January 2019
Available online xxxx

Keywords:
Text representation
Document classification
Feature extraction
Arabic document
Bi-gram alphabet
Support vector machine

ABSTRACT

In text classification, texts have to be transformed into numeric representations suitable for the learning algorithms. A main problem with the commonly used bag of words method is the high dimensions of vector space, as well as the need for language-dependent tools. In the present study, text classification is performed based on a novel bi-gram alphabet approach to construct feature terms. The proposed approach has two main contributions to text classification area. First, we have demonstrated the possibility of using constant feature terms that are based on the standard alphabet without the need for the documents vocabularies; this definitely helps in reducing the dimensions of the vector space for large corpus. Second, it does not require natural language processing tools. The current work has proved the ability to classify collections of Arabic or English text documents successfully. It showed approximately 80% savings in vector space and 2% performance improvement compared to the best recorded results on Arabic dataset Aljazeera News.

© 2019 The Author. Production and hosting by Elsevier B.V. on behalf of King Saud University. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

1. Introduction

Text classification/categorization is one of the challenging research topics due to the necessity to organize and categorize the growing number of electronic text documents on the Internet. It has been successfully applied to wide variety of domains such as news categorization, spam e-mail filtering, opinion mining, information retrieval, automatic indexing, summarization, and others. Text classification can be defined as the task of assigning natural language texts into one or more predefined categories based on their content (Sebastiani, 2002). While many researchers apply various machine learning algorithms to study the effectiveness of text classifiers, few works have considered exploring and analyzing possible ways of text representation schemes and their impact on the accuracy of text classification systems. Vast majority of researches utilize the vector space model VSM (Salton et al., 1975) that makes use of the bag of words BOW approach (Joachims, 1998a,b). In BOW method a document is converted into a vector of weighted words. First, all the words that occur in the corpus are defined. Then each text document is mapped into its feature vector based on the occurrences of the words in that document. The complete set of vectors for all documents under consideration yields a VSM. The length of a document vector is equal to the number of different words exist in all of the documents.

A main problem with BOW method is the high dimensions of the input feature space and the huge sparse matrix which in turn affects the performance and the complexity of the learning algorithms. For a big collection, the dimension of the bag-of-words vector space can reach hundreds of thousands. Therefore, to reduce the size of the vocabulary and improve the classification accuracy, two techniques are usually implemented by most of the researchers as a preprocessing task. First, removal of the words that is frequently used and common among all documents (so-called stop words) such as pronouns, propositions, articles, and conjunctions, etc. Second, converting all the words into their canonical forms (such as stem or root) and drop the repetitive ones. This helps to reduce the dimensionality of the feature space and the memory requirements. However, the preprocessing is further challenging especially in the case of high morphological languages including Arabic. Therefore, finding a way to perform successful classification without the need of language-dependent tools is a major contribution to TC area.

Arabic belongs to the Semitic languages family. It is the language of the Quran, the sacred book of Muslims. Around 250 million people use it as their first language. Arabic alphabet has 28 letters/characters and unlike English is written from right to left. In this work, the words *letter* and *character* are used interchangeably. There exist several types of short vowels which can be used

Peer review under responsibility of King Saud University.

Production and hosting by Elsevier

E-mail address: elghannam@eri.sci.eg

https://doi.org/10.1016/j.jksuci.2019.01.005
1319-1578/© 2019 The Author. Production and hosting by Elsevier B.V. on behalf of King Saud University.
This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

Please cite this article as: F. Elghannam, Text representation and classification based on bi-gram alphabet, Journal of King Saud University – Computer and Information Sciences, https://doi.org/10.1016/j.jksuci.2019.01.005

in generating different pronunciation of a letter. Arabic has rich morphology, both derivational and inflectional. It has a two-gender system, masculine or feminine and three verb forms: past, present and imperative. One of the main distinguishing features – as other Semitic languages- is the root-and-pattern morphology. Each word in the language is a combination of a set of base letters (root) plus a given pattern. Due to the fact that the modern standard Arabic publications do usually not encode short vowels and omits some other important phonological distinctions, the degree of morphological ambiguity is very high. In addition to this complexity, Arabic orthography prescribes to concatenate certain word forms with the preceding or the following ones, possibly changing their spelling and not just leaving out the whitespace in between them (Smrz, 2007). This convention complicates language processing to tokenize the distinct words, because they may be combined into one compact string of letters. Therefore, the ambiguity of the Arabic language, whether due its complex morphological nature or lack of diacritization, makes the pre-processing stage more complex than the English case. So, if there exist a method to perform successfully the classification process without the need of NLP tools, this will be a major advantage to TC area.

The traditional view of the relation between sound of a word and its meaning is arbitrary, i.e word meaning unaffected by its sound. An alternative hypothesis, known as sound symbolism that invokes non-arbitrariness refers to the apparent association between particular sound sequences and particular meanings in speech (Sapir, 1929; Nuckolls, 1999). The early study of Ibn Jinni (died at 1001) expressed this innate linguistic theory with his famous saying: "Creation of letters on the intended meaning and purpose" (Abbas, 1998). Abbas follows the sound symbolism approach and agree that "the meaning of the Arabic letter is the echo of its voice in the conscience, or soul". His study on the characteristics and meanings of Arabic letters had proved 50–91% compatibility of Arabic letters with their meanings in 3523 roots.

The current study is motivated by the notion of sound symbolism. We designed to investigate the possibility to extend the sound symbolism notion to text classification area by detecting the distribution of sequences of alphabetic letters to predict the document class. In this concern, we conducted a preliminary experiment based on unigram alphabet. The classification accuracy of that experiment was reasonable. This encourage us to extend using unigram to adopt the bigram alphabet.

In this paper, we present a novel approach to represent the text documents for TC. All possible bi-gram arrangements of the standard alphabet are used in constructing the features. Term frequency of bi-gram alphabet was used as a weighting scheme to represent document contents. A main contribution of this work to text classification area is the use of constant predefined standard feature terms instead of relying on documents vocabularies to extract the features. This guarantee reducing the dramatically increase in document vector space with increasing volume of data; thus helping to improve the complexity and performance of learning algorithms. Furthermore, the proposed approach avoids the use of tokenizers, stemmers or other language-dependent tools which are complex and may bring noise to representation especially for the high morphological languages including Arabic. The current work was applied to classify Arabic and English text collections, in addition to extensive study on Arabic text classification.

The proposed approach is characterized by:

- language-independent
- Easy to apply and does not require NLP tools.
- Construction of the feature terms is based on the standard alphabet.
- The feature terms are constant, and do not influenced by the vocabularies or genre of the corpus.
- Number of feature terms does not increase with increasing volume of data.
- Term frequency of bi-gram alphabet is used in constructing document vector.
- Train and test data divisions do not influenced by feature extraction or document vector calculations.

The rest of the paper is organized as follows: Section 2 introduces existing related work. Section 3 presents the proposed approach to classify documents. Experimental results and analysis are reported in Section 4, and the last section concludes this paper.

2. Related work

Text classification is one of the important research issues in the field of text mining. Text representation is a critical task that has a direct effect on the classification accuracy. Because text cannot be directly interpreted by a classifier algorithm, it needs to be mapped into a vector of numeric weights based on the documents contents. The following presents an overview of text classification regarding different text representation methods.

Most of text classification researches concentrate on BOW method, where each feature corresponds to a single word found in the training corpus. And so, all the words occurring in the corpus contribute to the feature vector configuration. To reduce the dimension of feature space, stem or root is used by many researchers (Syiam et al., 2006; Zahran et al., 2009; Bahassine et al. 2017), where different forms of the same word are consolidated into a single word. Feature selection is also a common technique in order to reduce irrelevant or noisy terms with the aim of improving the learning performance and saving the computation requirements (Guyon and Elisseeff, 2003). Various dimensionality reduction approaches have been conducted by the use of feature selection techniques, such as Information Gain, Mutual Information, Chi-Square, and so on. Details of these selection functions were stated in Nigam (2001).

Another approach is to apply a summarizer to extract informative sentences first, and select the best words from the summary documents instead of using all words. Ker and Chen (2000) applied the combination of word-based frequency and position method on news corpus to get categorization knowledge from the title field only. Their results indicate that summarization-based classification can achieve acceptable performance and a short computation time. Al-Thwaib (2014) applied the summarized Arabic data set as input for SVM classifier. His results showed an improvement using the words extracted from summary compared to using all words in feature representation.

In order to improve the classification accuracy, other works expand the representation with additional information, such as word n-grams and semantic knowledge. Word n-grams can be used to standalone as feature terms or in combination with unigrams (single words). The experimental results by (Bekkerman and Allan, 2004) showed that using only phrases as feature terms leads to a decrease in classification accuracy compared with the BOW baseline. Fürnkranz (1998) have used combinations of unigrams and n-grams as feature terms. Their classification results indicated that word sequences of length 2 or 3 usually improve classification performance, while longer sequences are not as useful. Even though N-grams are more representative than single words only, they are so sparsely distributed, and in turn it rises the problem of dimensionality. Because trigrams are sparser than bigrams, most of the researches have focused on the combination of unigrams and bigrams using different feature selection techniques. The work of Al-Shalabi and Obeidat (2008) apply the same representation method to classify Arabic documents. They used

Please cite this article as: F. Elghannam, Text representation and classification based on bi-gram alphabet, Journal of King Saud University – Computer and Information Sciences, https://doi.org/10.1016/j.jksuci.2019.01.005

unigrams and bigrams word level, and compared the classification results with that of traditional BOW. Their results showed that using N-grams produces better accuracy than using single words only in feature representation for the classification.

Another approach that is richer than simple BOW is to use ontology for document representation. The ontology model preserves the domain knowledge of a term present in a document. However, automatic ontology construction is still a difficult task due to the lack of structured knowledge base (Harish et al., 2010; Zhang et al., 2015). Bloehdorn and Hotho (2004) have proposed an approach that incorporate concepts from background knowledge into document representations for text document classification. Their experiments showed that the integration of concepts into the feature representation improves classification results. The work of Yousif et al. (2017) proposed an approach for Arabic TC using Arabic WordNet AWN thesaurus as a lexical and semantic source. They proposed a weighting scheme based on the frequency of the relation in the AWN and the corpus documents. The classification results showed that their suggested approach outperformed the BOW approach.

While the above-mentioned methods have focused on words to represent documents, other methods have used alternative approach where they dealt with characters to cope with the morphological processing. Kanaris et al. (2007) have proposed character-level n-grams with linear classifiers in the framework of content-based anti-spam filtering. Their results showed that character n-grams are more reliable features than word-tokens despite the fact that they increase the problem of dimensionality. A representation based on the combination of character bi-grams and tri-grams that are extracted from the documents is proposed in (Berger et al., 2005). The work of Santos and Zadrozny (2014) proposed a deep neural network for part of speech POS tagging. Their approach used words as a basis, in which character-level features extracted at word level form a distributed representation. Khreisat (2006) proposed an Arabic text classification using character tri-gram frequency. The distance between each document and all other documents is measured based on the tri-gram frequency profile. She compared the two distance measures, Manhattan and Dice. Sawaf et al. (2001) proposed a character based classifier to cope with the sparse data problem instead of feature reduction by the morphological rules. They extracted from the texts continuous sequences of two types of units: full-form words or character tri-grams. They carried out experiments with maximum entropy text classification on a large Arabic corpus and used no preprocessing steps.

3. Text classification using bi-gram alphabet approach

The proposed approach for text classification starts first by constructing the features terms, which are based only on bigrams of the standard alphabet. Then simple preprocessing which is only limited to orthographic normalization for Arabic documents is applied, while no preprocessing is applied for English documents. For each document, document vector is calculated based on occurrences of the bi-gram terms in that document. Finally k-fold cross-validation technique is applied and the popular SVM classifier is used to classify the documents. The detailed steps of the current approach are presented in the following sections.

3.1. Features construction

The current document classification approach adopted the standard alphabet to construct the feature terms. The generated features are standard and do not influenced by the corpus genre or content words. Letters are the basic seed to create the feature terms. Each alphabet letter is called a gram; a term of two-letters is, in turn, called bigram. To construct the feature terms, all possible arrangements of two alphabet letters for the specific language are generated, for example aa, ab, ac, … zy, zz.

An arrangement of a set of objects is called a permutation. The number of generated terms is equal to the permutation *P* for the number of distinct standard alphabet letters. The permutations of a number of ordered arrangements with repetitions of *r* objects taken from *n* unlike objects is defined by Permutation (2018):

$$^{n}P_{r} = n^{r} \tag{1}$$

where

- *r* is the number of letters chosen to construct a term.
- *n* is the number of distinct standard alphabet letters.

r = 2 for bi-gram term, while the number of letters in the alphabet depends on which language's alphabet used.

English alphabet contains 26 letters range from ‘*a*’ to ‘*z*’, i.e. *n = 26*.

So, total number of adopted English bi-gram letters arrangements (the repetitions of letters was allowed) is 676 terms, which represents the number of constructed features for the English documents.

The basic Arabic alphabet contains 28 letters range from ‘ا’ *Alif* to ‘ي’ *Ya*.

In the current work, when constructing the features for Arabic, we excluded double letters from appearing in the list of bigrams (ex: “تت“،”ققـ”) to reduce the feature space. Whereas preliminary experiments showed that their elimination has no effective impact on the performance of the classification process. In addition to, double letter cannot be detected directly in case of the same letter occurs twice in a word, with no vowel between. In this case, the letter is written only once and a short vowel “shadda” (or Madda in case the letter is Alif) is placed on top of it, which does not often appear in modern standard Arabic documents as mentioned in Section 1.

So, for the Arabic language the permutation P of a number of ordered arrangements *-with no repetitions-* of r objects taken from *n* unlike objects is defined by Permutations (2018):

$$\begin{aligned} &^{n}P_{r} = \tfrac{n!}{(n-r)!} \\ &r = 2,\ n = 28 \end{aligned} \tag{2}$$

Therefore, total number of Arabic bi-gram letters arrangements is 756, which represents the number of constructed features for Arabic documents.

It should be noted that, in addition to the benefits of using a limited number of features, the process of constructing the features itself in the presented approach as compared to BOW method saves a lot of tasks such as tokenization, and text cleaning.

3.2. Text Pre-processing

In Arabic some characters have different shapes due to several sounds it represents. For example, Alif, the first letter in Arabic alphabet and the most used letter among them due to the several sounds it represents, it has different shapes (إ ,آ ,أ ,ا) to recognize each sound. *Ha* (the counterpart of the English letter H) has different and unique shapes. The final shape of Ha (ـه) looks exactly like the feminine glyph Ta Marbootah (ـة). In the present work, preprocessing step is only limited to orthographic normalization of Arabic character shapes. Different aleph shapes “إ ,آ ,أ ,ا ” were normalized to “ا“ , and Ha shapes “ـه ، ـة” to “ـه”. However, in our side experiments it was found that orthographic normalization of the Arabic characters has no significant influence on the

Please cite this article as: F. Elghannam, Text representation and classification based on bi-gram alphabet, Journal of King Saud University – Computer and Information Sciences, https://doi.org/10.1016/j.jksuci.2019.01.005

performance of the text classifier. This is -in our opinion- due to the fact that these letters are generally much less than those in the corpus as a whole. Thus, the normalization step can be bypassed in the incoming works.

It should be noted that many of the preprocessing steps that are commonly used in BOW model, including cleaning of the functional words and reducing words to their stem or root, are deliberately avoided. The aim is to assess and examine accuracy of the current approach to classify documents using the simplest steps. As the preprocessing depends heavily on the language used and its structure, and therefore requires different NLP tools for each language. Thus, bypassing these steps while maintaining good performance is a significant benefit to the classification process.

For the English documents no preprocessing step was applied at all.

3.3. Document vector calculation

The next step is to score each document in the corpus. The objective is to turn each document of free text into a numerical vector that can be used as input for a machine learning model. Document vectors are derived from the textual data and the feature terms, they describe the occurrence of the bi-gram feature terms within documents, considering each bigram count as a feature. There are various presentations or weights methods used in the literature of TC, we choose the simple term frequency adjusted for document length which is called normalized term frequency *ntf(t,d*). term frequency *tf(t,d)* is the number of times a term *t* occurs in a document *d*. normalized term frequency *ntf(t,d)* is the number of times a term occurs in a specific document *d*, normalized by the total number of terms exist in that document. *ntf(t,d)* values ranges from 0 to 1.

$$ntf(t,d) = tf(t,d) \div number\ of\ terms\ in\ d \tag{3}$$

It is clear that the length of a document vector is equal to the total number of constructed feature terms depending on the language used.

3.4. Feature selection

Not all combinations of two alphabet characters are common in language; certainly there are some rare compositions. The presence of others does not have an effective impact on the classification process. Hence, to reduce the dimensionality of vector space and the existence of irrelevant (noisy) features, it was desirable to apply feature selection. The feature selection stage ensures that those features which are highly skewed towards the presence of a particular class label are picked for the learning process. One of the most common methods for feature selection used in the literature of TC is the chi-square $\chi 2$ statistic. It has proved high accuracy in classifying both Arabic and English texts (Joachims, 1998a,b; Anitha et al., 2013; Al-Tahrawi and Al-Khatib, 2015). The $\chi 2$ statistic computes the lack of independence between the term t and a particular class *i*. Let *n* be the total number of documents in the collection, *pi(t)* be the conditional probability of class *i* for documents which contain t, Pi be the global fraction of documents containing the class *i*, and *F(t)* be the global fraction of documents which contain the term t. The $\chi 2$-statistic of the term between term *t* and class *i* is defined as follows (Aggarwal and Zhai, 2012):

$$\chi^{2i(t)} = \frac{n \cdot F(t)2 \cdot (pi(t) - Pi)2}{F(t) \cdot (1 - F(t)) \cdot (Pi \cdot (1 - Pi))} \tag{4}$$

So, $\chi 2$-statistic was used in the current work to determine the most discriminating features for Arabic TC; the top p percent features were selected to build the classifier. Results of different *p* values were tested during the experiments in Sections 4.2, 4.4, 4.5.

3.5. Machine learning process

After constructing the document vectors, the phase of choosing appropriate classifier can be applied. The goal is to find the algorithm that achieves close to maximum accuracy while minimizing computation time required for training. There is a number of successful classifiers that have been used in text classification. SVM has proven that it is very well suiting for text classification for its high accuracy, and an inherent ability to handle large feature spaces (Anitha et al., 2013; Cristianini and Shawe-Taylor, 2000; Joachims, 1998a,b). The main principle of SVMs is to determine separators in the search space which can best separate the different classes. The separation process depends on the maximum distance between the two sides of hyper plane and the nearest vectors in the training data sample. One advantage of SVM is that since it attempts to determine the optimum direction of discrimination in the feature space by examining the appropriate combination of features, it is quite robust to high dimensionality (Aggarwal and Zhai, 2012).

Training a support vector machine requires the solution of a very large quadratic programming (QP) optimization problem. SVM classifier finds an optimal separating hyperplane that maximizing the distance of either class to the separating hyperplane, and at the same time minimizing the risk of misclassifying the training data. Sequential minimal optimization (SMO) is an algorithm for solving the optimization problem that arises during the training of support vector machines. It breaks this problem into a series of smallest possible sub-problems, which are then solved analytically. SMO; which is WEKA (Witten and Frank, 2005) version of SVM algorithm was used to build the model. In our experiments, the dataset was tested using k-fold cross-validation technique with k = 10. In this technique, the original dataset is randomly partitioned into k subsamples. A single subsample is retained as the validation data for testing the model, and the remaining k − 1 subsamples are used as training data. The process is repeated k times, with each of the k subsamples used exactly once as the test data. The k results then can be averaged to produce the final estimation.

It is noted that, in the current work we do not need to do feature extraction or term weighting separately for the training set and the testing set. Where the feature terms are standard and do not extracted from the documents contents. At the same time, term weighting calculation for a document, is not influenced by other documents. Other works that have used BOW and TF.IDF need to do feature preprocessing and weightings separately for the training set and the test set.

4. Experimental study

We have conducted experiments to classify text collections using the proposed approach on four Arabic datasets and four English datasets. Nonetheless, we gave more study on Arabic TC. In the experiments, precision, recall and F-measure were used as a performance metrics for TC. In all experiments conducted, the classifier was trained with the 10-fold cross validation technique. Four different experiments were conducted. Both Weka (Witten et al., 2016) and Rapid Miner (Rapid Miner Project, 2013) tools were used during the experiments. The datasets used, results of experiments conducted, performance evaluation, details of feature reduction, and analysis of results are presented and discussed in the following sections.

4.1. The datasets

In our, eight different Arabic and English datasets were used as seen in Tables 1 and 2.

Table 1
Arabic datasets used in the experiments.

Dataset	Number of documents	Number of classes
Alkhaleej news	2000	4
Alj-News9	2700	9
Alj-News5	1500	5
BBC-Arabic news	4763	7

Table 2
English datasets used in the experiments.

Dataset	Number of documents	Number of classes
BBC-English news	2225	5
Reuters R8	7674	8
20Ng	18,821	20
Subjectivity	10,000	2

4.1.1. Arabic datasets

4.1.1.1. Alkhaleej news dataset. Alkhaleej-2004 dataset was prepared by Mourad Abbas (Arabic Corpora – Mourad Abbas, 2004). The dataset contains 5690 documents which correspond to nearly 3 million words.

Each document is labeled with one of the following four classes {'International News', 'Local News', 'Sport', and 'Economy'}. We randomly selected a set of 2000 documents equally distributed among the four classes.

4.1.1.2. Aljazeera news 9 classes dataset (Alj-News9). Alj-News9 Arabic Dataset contains 2700 documents for news articles (Arabic Corpora – Alj-News, 2004). Each document is labeled with one of the following nine classes {'Economy', 'Health', 'Law', 'Literature', 'Politics', 'Religion', 'Sport', 'Technology', and 'Art'} with equal number of documents in each category.

4.1.1.3. Aljazeera news 5 classes dataset (Alj-News5). Alj-News5 is another different dataset that contains 1500 documents for news articles (Arabic Corpora – Alj-News, 2004). Each document is labeled with one of the following five classes {'Art', 'Economic', 'Politics', 'Science', and 'Sport'}. Alj-News5 was used by other researchers (Chantar and Corne, 2011; Al-Tahrawi and Al-Khatib, 2015). A comparison of classification results of the current approach against other research works on Alj-News5 will be presented in the experiments.

4.1.1.4. BBC-Arabic news dataset. The dataset contains 4763 documents of BBC-Arabic news (Saad and Ashour, 2010) collected from the BBC Arabic website. Each document is labeled with one of the following seven classes {'Middle East', 'World News', 'business', 'sport', 'newspapers', 'Science', and 'Misc.'}.

4.1.2. English datasets

4.1.2.1. BBC-English news dataset. The dataset contains 2225 articles from the BBC news website (Greene and Cunningham, 2006) corresponding to stories in five topical areas from years 2004–2005. Each article is labeled with one of the following five classes: {'business', 'entertainment', 'politics', 'sport', and 'tech'}.

4.1.2.2. Reuters-21578 dataset (Reuters R8). The dataset appeared on the Reuters newswire in 1987 and were manually classified by personnel from Reuters Ltd. Due to the fact that the class distribution for these documents is very skewed, two sub-collections R10 and R90 (contain 10 and 90 classes respectively) are usually considered for text categorization tasks. Cachopo (2007) identified 8 of the 10 most frequent classes on R10 to prepare R8. Only the documents with a single topic are considered in this version. We use R8 version, it contains 7674 documents distributed across 8 classes {Acq, crude, earn, grain, interest, money-fx, ship, trade}.

4.1.2.3. 20 Newsgroups dataset (20Ng). The original dataset is a collection of approximately 20,000 newsgroup documents, partitioned evenly across 20 different newsgroups. Cachopo (2007) prepared the original dataset by removing some duplicates, empty messages, and PGP keys. In our experiments, Cachopo version was used. It contains 18,821 documents distributed across 20 classes.

4.1.2.4. Subjectivity dataset. The dataset contains 10,000 reviews snippets for two movies from the Internet Movie Database IMDb (Rotten Tomatoes and Plot Summaries movies). They assumed that all snippets from the Rotten Tomatoes pages are subjective, and all sentences from IMDb plot summaries are objective. This is mostly true; but plot summaries can occasionally contain subjective sentences that are mis-labeled as objective (Pang and Lee, 2004).

4.2. Bi-gram alphabet distribution in Arabic vocabularies

The experiment was conducted to study the distribution of bi-gram alphabet in Arabic vocabularies. In this experiment Chi-square was applied to measure the importance of each feature. The experiment was applied to classify documents using the four Arabic datasets separately.

In studying the results of Chi-square term weights, it was found that there are typical 125 terms that have zero weights agreed upon all the four Arabic datasets. This represents 16% of the total number of Arabic feature terms. It was observed that those bi-gram terms almost do not occur in the datasets vocabularies. We therefore considered these terms as rare and difficult to occur in the Arabic vocabularies. Accordingly, the list of rare bi-gram terms can be safely dropped from the feature list as it will be shown in next experiment. Example of zero weight rare bi-grams in Arabic vocabularies that were revealed by this experiment are: (ذش ,ضظ ,صض ,شص ,سش , دذ ,حخ).

4.3. Validation of bi-gram alphabet approach

In this experiment we separately classified different Arabic and English datasets using SVM-SMO classifier and bi-gram alphabet vector representation. The experiment aims to find out whether the approach is applicable to classify Arabic or English documents. The classifier was trained with 10-fold cross validation technique. For the Arabic documents, the experiment was applied using 631 features (full number of features excluding the zero weight rare ones described in Section 4.2. Concerning the English documents, 677 feature terms were used (full features) in the experiment. Tables 3 and 4 show the overall summarized classification accuracy in terms of Precision, Recall, and F-measure on different Arabic and English datasets. In Table 3 it can be seen that F-measure results fall in range between (0.949) and (0.874) on the four Arabic datasets, Alj-News5 dataset has the highest F-measure while BBC-Arabic dataset has the lowest. In Table 4 it can be seen that F-measure results fall in range between (0.937) and (0.710) on the four English datasets, Reuters R8 dataset has the highest F-

Table 3
Classification accuracy of SVM-SMO on different Arabic datasets.

Dataset	Precision	Recall	F-measure
Alkhaleej	0.905	0.905	0.905
Alj-News9	0.930	0.930	0.930
Alj-News5	0.949	0.949	0.949
BBC-Arabic	0.874	0.874	0.874

Table 4
Classification accuracy of SVM-SMO on different English datasets.

Dataset	Precision	Recall	F-measure
BBC-English	0.926	0.926	0.926
Reuters R8	0.938	0.937	0.937
20Ng	0.711	0. 710	0.710
Subjectively	0.811	0.811	0.811

measure while 20Ng dataset has the lowest. There are a number of factors related to the dataset that affect the classification accuracy. This includes dataset genre, number of instances and classes, distribution of documents along existing classes, existence of noisy data, and the interference between classes.

More detailed study and comparison against other works in Arabic text classification will be presented in the following experiments.

4.4. Effect of feature space reduction on the classification accuracy

The experiments were conducted with different feature space to study the effect of selected number of features on the achieved accuracy. Initially, features were weighted by Chi-square and ordered by their weights in ascending order. Then feature dimension was defined by 100%, 90% and so on up to 10% of full features (where full features = 756 in Arabic). The classifier was trained with 10-fold cross validation technique on different Arabic datasets separately. Table 5 shows the detailed view of accuracy results among different percentage of highest ranking features with Chi-square feature selector. The columns of the table indicate the percentage features selection, the corresponding number of features selected, and the recorded F-measures for different Arabic datasets. Table 6 illustrates values of the average deviation from the baseline at different percentages of highest ranking features. The classification accuracy results in terms of F-measure were formulated in the two graphs shown in Figs. 1 and 2. The figures illustrate the effect of selected percentage number of features with Chi-square feature selector on the achieved accuracy over different Arabic data sets. Fig. 1 shows the accuracy results among different percentages of highest ranking features over different datasets. Fig. 2 shows the accuracy deviations from the baseline over different data sets with different percentages of the highest ranking features. Where the baseline is the accuracy at full feature.

It is clear from Table 5 that for all datasets the best accuracy was achieved at full features. However, as it is clear from Table 6 that even the selected features of up to 50% of full features, there was no significant change in the accuracy results (less than 1%) over all datasets. At (40–20) % of full features, the average accuracy starts to decrease gradually by (1–3) % compared to the results at full accuracy. While it is observed a severe degradation in accuracy about (7%) when using 10% of full features.

Table 5
Different feature selections and corresponding F-measure on Arabic datasets.

% Feature selection	Number of features	Alkhaleej	Alj-News9	Alj-News5	BBC-Arabic
100%	756	0.905	0.930	0.949	0.874
90%	680	0.905	0.930	0.949	0.874
80%	605	0.905	0.930	0.949	0.874
70%	529	0.904	0.928	0.949	0.874
60%	454	0.903	0.926	0.944	0.872
50%	378	0.902	0.925	0.946	0.871
40%	302	0.900	0.921	0.940	0.867
30%	226	0.894	0.916	0.930	0.861
20%	151	0.875	0.900	0.914	0.846
10%	75	0.841	0.859	0.890	0.799

Table 6
Average deviation in accuracy from the baseline using different % feature selections.

% Feature selection	Average deviation
90%	0.000
80%	0.000
70%	0.001
60%	0.002
50%	0.003
40%	0.007
30%	0.014
20%	0.030
10%	0.067

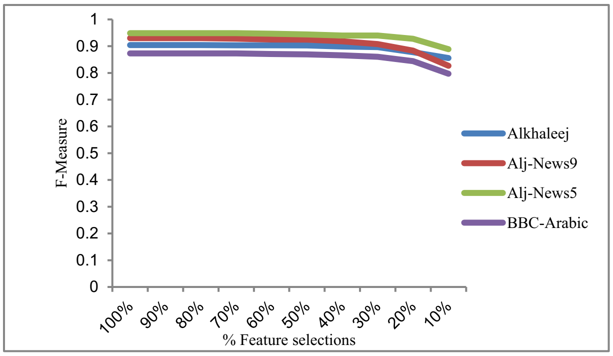

Fig. 1. % Features selection and the corresponding classification accuracy.

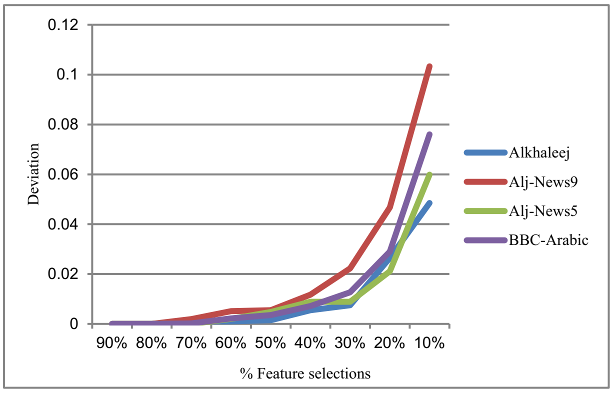

Fig. 2. Deviation in accuracy from the baseline using different % feature selections.

The results show that the proposed approach can be used with approximately the same full efficiency by considering only 50% of full features, i.e. 378 Arabic terms instead of 756. Although there is deterioration in the classifier efficiency at 10% compared to the full features case, the classifier still works with reasonable results. This indicates that there exist about (10–50)% of the bi-gram alphabet terms that play crucial role in document representation for the classification process. More details and comparison against other methods will be presented in the next experiment.

4.5. Analysis and comparison against other Arabic TC methods

The detailed classification results of Alj-News5 were highlighted in order to analyze the results more closely and compare the accuracy against other methods. There are difficulties to make direct comparison with other previous research works in the area of Arabic TC. Among these difficulties are the following: no

Please cite this article as: F. Elghannam, Text representation and classification based on bi-gram alphabet, Journal of King Saud University – Computer and Information Sciences, https://doi.org/10.1016/j.jksuci.2019.01.005

benchmark Arabic dataset, the dataset used is not available in some cases, researchers select randomly different amount of documents of the original dataset, different number of instances for each class, different organization of training and testing sets, and lack of clarity on the number of features selected. Therefore, Alj-News5 was selected for comparison due to the availability of its test results by other works.

The detailed results of applying SVM-SMO classifier using the current bi-gram alphabet approach on Alj-News5 are presented in Table 7 showing Precision, Recall, and F-measure for each class. The classifier was trained with the 10-fold cross validation technique as described in Section 4.3. As shown in Table 7, the top F-measure was on 'Sport' class with 0.995, while the lowest was 0.901 on 'Politics' class. The overall weighted average F-measure for all classes was 0.949. Table 8 shows the corresponding confusion matrix, which indicates labels that were correctly predicted, and the off-diagonal entries indicate errors, for example the substitution of 16 labels of 'art' for 'politics'.

In order to measure the success of the proposed approach, the results were compared against other two works; System1, proposed by Chantar and Corne (2011), and System2, proposed by Al-Tahrawi and Al-Khatib (2015), for the same dataset Alj-News5. Due to the important impact of the dimension of feature space in the classification process, it was taken into consideration in the comparison. Table 9 presents the best overall F-measure and the number of features selected for the three systems on Alj-News5 dataset.

Despite the difficulty of an accurate comparison, the analysis of the results showed the following:

System1 used BPSO-KNN (Binary Particle Swarm Optimization)-KNN) as a feature selection method. They applied preprocessing steps that include removing stop words and rare words, no stemming was applied. They experiment different classifiers, their best results on Alj-News Arabic corpus (Alj-News5) was (F-measure = 0.931) obtained by SVM classifier. Nevertheless, this accuracy value was at the expense of high memory requirements due to the large number of features (2967) that was used to build the classifier. The results obtained by the current work outperformed system1 in terms of f-measure (0.949), and also for the number of features used (5 2 9). In addition to, in the current approach, to achieve approximately the same accuracy (0.930) that was obtained by System1, only 266 features could be used, as shown in Table 5.

Table 7
Accuracy by class for SVM-SMO on Alj-News5.

Class	Precision	Recall	F-Measure
Art	0.937	0.937	0.937
Economic	0.946	0.930	0.938
Politics	0.890	0.913	0.901
Science	0.980	0.967	0.973
Sport	0.993	0.997	0.995
Weighted Avg.	0.949	0.949	0.949

Table 8
Confusion matrix for Alj-News5.

A	B	C	D	E	Classified as
281	1	16	1	1	A = art
2	279	16	3	0	B = economic
14	9	274	2	1	C = politics
2	6	2	290	0	D = science
1	0	0	0	299	E = sport

Table 9
Overall best F-measure and number of features selected for the three systems on Alj-News5 dataset.

Work	F-measure	Number of features
System1	0.931	2967
System2	0.893	135
Bi-gram alphabet	0.949	529

System2 They applied several preprocessing steps including stop words removal and stemming. PN classifier was used to classify Arabic documents. In their work, to further reduce the number of features, Chi-square was used for feature selection, and only 1% of each class features was selected to build the classifier. Their results on Alj-News Arabic corpus (Alj-News5) using 135 features was (F-measure = 0.893). The current approach has the superiority in f-measure (0.949). Although System 2 have used a powerful algorithm PNs classifier, it has memory restriction. So, there was a need to reduce the number of features as have been implemented in their work. In this regard, another experiment was applied to test the current bi-gram alphabet approach using the same number of features that was used in system2. The results showed that bi-gram alphabet achieved 0.913 at 135 selected features, compared to 0.893 that was obtained by system2 at the same number of features.

In addition to that the proposed approach had proved high accuracy results; there are two basic remarkable contributions in the area. First, features that were used in the classification process are standard and separate from contents of the documents, so it does not increase with increasing volume of data; this is an important issue to keep the memory requirements at minimum. Second, the results were obtained without the need of complex NLP tools. So we have escaped a difficult task for several languages, in particular for the high inflectional languages including Arabic.

5. Conclusion

In this work, we designed a novel bigram alphabet approach for features construction and its application in text classification area. Term frequency of bi-gram alphabet was used as a weighting scheme to represent document contents. The approach is language independent and does not require NLP tools. Using SVM-SMO classifier, the proposed approach has proved the ability to classify collections of Arabic or English text documents successfully. The results on Arabic datasets show that the proposed approach can be used with approximately the same full efficiency by considering only 50% of the bigram alphabet terms i.e. 378 features. The current approach has two main contributions. First, features are standard and separate from contents of the documents; this helps to reduce the high dimensionality with increasing the volume of data. Second, the classification process can be performed without the need for NLP tools which are complex, especially for high morphological languages including Arabic.

References

Abbas, H. (1998). العرب خصائص الحروف العربية ومعانيها: دراسة. منشورات اتحاد الكتاب.

Aggarwal, C.C., Zhai, C. (Eds.), 2012. Mining text data. Springer Science & Business Media.

Al-Shalabi, R., Obeidat, R., 2008. Improving KNN Arabic text classification with n-grams based document indexing. In: Proceedings of the Sixth International Conference on Informatics and Systems, pp. 108–112.

Al-Tahrawi, M.M., Al-Khatib, S.N., 2015. Arabic text classification using Polynomial Networks. J. King Saud Univ.-Comput. Inf. Sci. 27 (4), 437–449.

Al-Thwaib, E., 2014. Text summarization as feature selection for Arabic text classification. World of Comput. Sci. Inf. Technol. J. (WCSIT) 4 (7), 101–104.

Please cite this article as: F. Elghannam, Text representation and classification based on bi-gram alphabet, Journal of King Saud University – Computer and Information Sciences, https://doi.org/10.1016/j.jksuci.2019.01.005

Anitha, N., Anitha, B., Pradeepa, S., 2013. Sentiment classification approaches. Int. J. Innovat. Eng. Technol. 3 (1), 22–31.

Arabic Corpora – Alj-News, 2004. Retrieved September 07, 2016, from https://filebox.vt.edu/users/dsaid/Alj-News.tar.gz. Last access on January 2013.

Arabic Corpora – Mourad Abbas, 2004. Retrieved September 04, 2018, from https://sites.google.com/site/mouradabbas9/corpora. Last access on January 2018.

Bahassine, S., Madani, A., Kissi, M., 2017. Arabic text classification using new stemmer for feature selection and decision trees. J. Eng. Sci. Technol. 12 (6), 1475–1487.

Bekkerman, R., Allan, J., 2004. Using bigrams in text categorization. Technical Report IR-408, Center of Intelligent. Information Retrieval, UMass Amherst.

Berger, H., Köhle, M., Merkl, D., 2005. On the Impact of Document Representation on Classifier Per-formance in e-Mail Categorization. In: ISTA (pp. 19–30).

Bloehdorn, S., Hotho, A., 2004. Boosting for text classification with semantic features. In International workshop on knowledge discovery on the web (pp. 149–166). Springer, Berlin, Heidelberg.

Cachopo, A.M.D.J.C., 2007. Improving methods for single-label text categorization. Instituto Superior Técnico, Portugal.

Chantar, H.K., Corne, D.W., 2011. Feature subset selection for Arabic document categorization using BPSO-KNN. In: 2011 Third World Congress on Nature and Biologically Inspired Computing (NaBIC). IEEE, pp. 546–551.

Cristianini, N., Shawe-Taylor, J., 2000. An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods. Cambridge University Press.

Fürnkranz, J., 1998. A study using n-gram features for text categorization. Austrian Res. Inst. Artif. Intell. 3 (1998), 1–10.

Greene, D., Cunningham, P., 2006. Practical solutions to the problem of diagonal dominance in kernel document clustering. In: Proceedings of the 23rd international conference on Machine learning. ACM, pp. 377–384.

Guyon, I., Elisseeff, A., 2003. An introduction to variable and feature selection. J. Mach. Learn Res. 3 (Mar), 1157–1182.

Harish, B.S., Guru, D.S., Manjunath, S., 2010. Representation and classification of text documents: a brief review. IJCA, Special Issue on RTIPPR 2, 110–119.

Joachims, T., 1998a. Text categorization with support vector machines: Learning with many relevant features. In: European Conference on Machine Learning. Springer, Berlin, Heidelberg, pp. 137–142.

Joachims, T., 1998b. Text categorization with support vector machines: Learning with many relevant features. In: European conference on machine learning. Springer, Berlin, Heidelberg, pp. 137–142.

Ker, S.J., Chen, J.N., 2000. A text categorization based on summarization technique. In: Proceedings of the ACL-2000 workshop on Recent advances in natural language processing and information retrieval: held in conjunction with the 38th Annual Meeting of the Association for Computational Linguistics-Volume 11. Association for Computational Linguistics, pp. 79–83.

Kanaris, I., Kanaris, K., Houvardas, I., Stamatatos, E., 2007. Words versus character n-grams for anti-spam filtering. Int. J. Artif. Intell. Tools 16 (06), 1047–1067.

Khreisat, L., 2006. Arabic text classification using N-gram frequency statistics a comparative study. In: Proceedings of the 2006 International Conference on Data Mining, pp. 78–82.

Nigam, K. P. (2001). Using unlabeled data to improve text classification. PhD Thesis, School of Computer Science, Carnegie Mellon University, USA.

Nuckolls, J.B., 1999. The case for sound symbolism. Annu. Rev. Anthropol. 28, 225–252.

Pang, B., Lee, L., 2004. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In: Proceedings of the 42nd annual meeting on Association for Computational Linguistics. Association for Computational Linguistics, p. 271.

Permutations: Retrieved October 10, 2018, from https://en.wikipedia.org/wiki/Permutation.

Rapid Miner Project RM, 2013. The Rapid Miner Project for Machine Learning. Available: http://rapid-i.com/ Last access on December 2017.

Saad, M.K., Ashour, W., 2010. Osac: Open source Arabic corpora. In 6th ArchEng Int. Symposiums, EEECS (Vol. 10).

Sapir, E., 1929. A study in phonetic symbolism. J. Exp. Psychol. 12, 239–255. https://doi.org/10.1037/h0070931.

Salton, G., Wong, A., Yang, C., 1975. A vector space model for automatic indexing. Commun. ACM 18 (11), 613–620.

Santos, C.D., Zadrozny, B., 2014. Learning character-level representations for part-of-speech tagging. In: Proceedings of the 31st International Conference on Machine Learning (ICML-14), pp. 1818–1826.

Sawaf, H., Zaplo, J., Ney, H., 2001. Statistical classification methods for Arabic news articles. Arabic Natural Language Processing Workshop, ACL'2001, pp. 127–132.

Sebastiani, F., 2002. Machine learning in automated text categorization. ACM Comput. Surveys (CSUR) 34 (1), 1–47.

Smrz, O., 2007. Functional Arabic Morphology. Formal system and Implementation PhD Thesis. Charles University, Prague, Czech Republic.

Syiam, M.M., Fayed, Z.T., Habib, M.B., 2006. An intelligent system for Arabic text categorization. Int. J. Intell. Comput. Inf. Sci. 6 (1), 1–19.

Witten, I.H., Frank, E., 2005. Data Mining: Practical Machine Learning Tools and Techniques. Morgan Kaufmann.

Witten, I.H., Frank, E., Hall, M.A., Pal, C.J., 2016. Data Mining: Practical Machine Learning Tools and Techniques. Morgan Kaufmann.

Yousif, S.A., Samawi, V.W., Elkabani, I., 2017. Arabic Text Classification: The Effect of the AWN Relations Weighting Scheme. Proceedings of the World Congress on Engineering.

Zahran, M.M., Kanaan, G., Habib, M.B., 2009. Text feature selection using particle Swarm optimization algorithm. World Appl. Sci. J. 7 (Special Issue of Computer, IT), 69–74.

Zhang, S., Boukamp, F., Teizer, J., 2015. Ontology-based semantic modeling of construction safety knowledge: Towards automated safety planning for job hazard analysis (JHA). Autom. Constr. 52, 29–41.