Arabic Text Classification using Bag-of-Concepts Representation

Alaa Alahmadi, Arash Joorabchi and Abdulhussain E. Mahdi
Electronic and Computer Engineering Department, University of Limerick, Limerick, Ireland

Keywords: Automatic Text Classification, Arabic Text, Bag-of-Words, Bag-of-Concepts, Wikipedia.

Abstract: With the exponential growth of Arabic text in digital form, the need for efficient organization, navigation and browsing of large amounts of documents in Arabic has increased. Text Classification (TC) is one of the important subfields of data mining. The Bag-of-Words (BOW) representation model, which is the traditional way to represent text for TC, only takes into account the frequency of term occurrence within a document. Therefore, it ignores important semantic relationships between terms and treats synonymous words independently. In order to address this problem, this paper describes the application of a Bag-of-Concepts (BOC) text representation model for Arabic text. The proposed model is based on utilizing the Arabic Wikipedia as a knowledge base for concept detection. The BOC model is used to generate a Vector Space Model, which in turn is fed into a classifier to categorize a collection of Arabic text documents. Two different machine-learning based classifiers have been deployed to evaluate the effectiveness of the proposed model in comparison to the traditional BOW model. The results of our experiment show that the proposed BOC model achieves an improved performance with respect to BOW in terms of classification accuracy.

1 INTRODUCTION

Arabic, which is the fifth most spoken language in the world, belongs to the Semitic group of languages and it is a highly inflectional and derivational language (Versteegh and Versteegh, 1997). The Arabic alphabet consists of 28 letters, and letters are directly connected when forming words. Arabic text is read from right to left. Unlike English, proper nouns do not start with capital letters which makes the process of recognizing and extracting them difficult. Moreover, 22 of the letters take different shapes based on their position in the word, i.e., initial, medial, or final. Table 1 shows the letter *b* (ب) appearing in different sample words and how its position affects its shape.

Table 1: Example of an Arabic letter and its various shapes depending on its position in the word.

position	word	Letter shape
Start	بدر	بـ
Middle	قبر	ـبـ
End	قلب	ب

The exponential growth of Arabic documents in digital form on the web has increased the need to assist users with the fast and effective navigation, browsing, and discovery of useful information on the Internet. Text Classification (TC) is the task of assigning one or more predefined categories to a given text.

Classification of Arabic text can be a challenging task due to the rich and complex nature of the Arabic language. The majority of reported works on Arabic TC attempt to represent text by using the Bag-of-Words (BOW) model. In this model, each document, $d_i \in D$, is expressed as a weighted high dimensional vector, $\vec{d}_i$, where each dimension corresponds to a unique word. For example, Alsaleem (2011) used Support Vector Machine (SVM) and Naïve Bayes (NB) classification algorithms using the BOW model to classify a Saudi Arabic newspaper text collection by Al-Harbi et al. (2008). The classification system yielded a Macro F1 score of 77.85% and 74.0% for SVM and NB respectively. Kanaan et al. (2009) used the BOW model for Arabic TC using a dataset compiled by Mesleh (2007) with different weighting schemes such as Term Frequency (TF), Term Frequency Inverse Document Frequency (TFIDF), and Weighted Inverse Document Frequency (WIDF).

Alahmadi A., Joorabchi A. and E. Mahdi A..
Arabic Text Classification using Bag-of-Concepts Representation.
DOI: 10.5220/0005138103740380
In *Proceedings of the International Conference on Knowledge Discovery and Information Retrieval* (KDIR-2014), pages 374-380
ISBN: 978-989-758-048-2

Performance comparisons between *k*-Nearest Neighbors (*k*-NN), Rocchio, and NB as classifiers were conducted. Results of this experiment show that the WIDF scheme yields the best performance when used in conjunction with the *k*-NN, while TFIDF shows the best performance when used in conjunction with Rocchio. Among the three classifiers, the NB classifier is reported to be the best performer yielding a Macro F1 score of 84.53%. Mesleh (2007) reported a BOW based Arabic TC system which uses Chi-square for feature selection. Their results show that using a SVM classifier in this context yields better classification performance compared to a *k*-NN or a NB classifier when features are reduced using Chi-square. It yields a Macro F1 score of 88.11%, when evaluated using their in-house compiled Arabic text dataset.

The BOW model suffers from two main limitations: (1) it breaks terms into their constituent words, e.g., it breaks 'text classification' into the words 'text' and 'classification'; as a result the order of the words is lost in the model and the meaning of the terms could be changed; (2) it treats synonymous words as independent features, e.g., 'classification' and 'categorization' are considered as two independent words with no semantic association. As a result of that documents which discuss similar topics and contain synonymous words could be considered unrelated.

Researchers have attempted to address the above issues in English TC by representing text as concepts rather than words, using an approach known as Bag-of-Concepts (BOC). A concept is a unit of knowledge with a unique meaning (ISO, 2009). To build a BOC model, semantic knowledge bases such as WordNet[1], Open Directory Project (ODP)[2], and Wikipedia[3] are used to identify the concepts appearing within a document. By using concepts in text representation the semantics and associations between words appearing in the document will be preserved. For example, Hotho et al (2003) used the English Wordnet as a knowledge base to represent English text. For each term in a document, Wordnet returns an ordered list of synonyms and the first ranked synonym will be used as a concept for the term. In that study, three approaches were proposed for using concepts as features for text representation; (1) using only concepts to represent documents; (2) Adding Concept (AC) as complimentary features to the BOW model ; (3) Replacing Term words with Concepts (RTC) in BOW model. The study shows that AC yields better classification performance results than the other two approaches. The results of this study show that representing documents with only concepts is insufficient as WordNet does not cover all special domain vocabularies. Furthermore, WordNet is limited as it is a manually constructed dictionary and therefore laborious to maintain.

To deal with this problem, other researchers have tried replacing WordNet with other knowledge bases derived from the Internet, such as ODP and Wikipedia, e.g., see (Gabrilovich and Markovitch, 2005, Gabrilovich and Markovitch, 2006). In these studies, ODP categories and Wikipedia articles are used as concepts for text representation. For each document, a text fragment (such as word, sentence, paragraph, or the whole document) maps to the most relevant ODP categories or Wikipedia articles. The mapped concepts are added to the document using the AC approach. Using these knowledge bases for BOC modelling improved the performance when applied to English TC compared to BOW model.

For Arabic TC, Elberrichi and Abidi (2012) used the Arabic WordNet (Black et al., 2006) to identify concepts appearing within the documents. A comparison between different text representation models such as BOW, N-grams and BOC was conducted using an Arabic text dataset collected by Mesleh (2007). An RTC variation of BOC used in conjunction with Chi-square for feature selection and a *k*-NN classifier was reported to achieve higher performance results compared to other representations.

In a previous work, we developed a number of new approaches, which combine the BOW and the BOC models, and applied them to English TC (Alahmadi et al., 2013) and Arabic TC (Alahmadi et al., 2014). In (Alahmadi et al., 2014), a NB classification algorithm was shown to provide better performance in conjunction with the BOC model compared to BOW model. This current work focuses on this point. The remainder of the paper is organized as follows: Section 2 describes why Wikipedia is a suitable knowledge base for BOC modelling of Arabic text. Section 3 outlines the pre-processing phase and describes the BOC model in details. The experimental set-up and results are discussed in Section 4. The paper concludes in Section 5.

2 WIKIPEDIA

Wikipedia is the largest electronic knowledge

[1] http://wordnet.princeton.edu/.

[2] http://dmoz.org.

[3] http://www.wikipedia.org.

repository on the Internet and its content is entirely contributed collaboratively by volunteers. Wikipedia is a comprehensive, up to date, and well-formed knowledge source.

In order to use Wikipedia as a knowledge source for building BOC models of Arabic text, an open-source toolkit known as Wikipedia-Miner (Milne and Witten, 2013) is utilized. First the toolkit processes Wikipedia XML dump files[4] and creates a database that contains a summarized version of Wikipedia's content and structure. The toolkit assigns each Wikipedia page a unique id, title, and type (article, category, or redirect). Each article in Wikipedia describes a single concept and the article title is the descriptor of the concept. The title is well formed, brief and can be used as a descriptor in ontologies. The aim of the redirect pages in Wikipedia is to connect articles with alternative titles that correspond to their synonyms.

The great majority of Wikipedia articles, i.e. concepts, are classified to one or more Wikipedia categories, which are hierarchical and descend from a single root. The maximum category depth in Arabic Wikipedia is 12. A concept can belong to multiple parent categories which are more general articles than the concept itself. In addition more specific concepts which are child articles can be mined by the concept as a parent category for them.

In Wikipedia, an article may have more than one redirect and other Wikipedia articles may link to it (link anchors). All these elements offer additional information, and they are grouped into labels. These labels will be used as concepts to represent text in different applications. In this study we have used Wikipedia labels to build the BOC representation model for Arabic TC.

3 ARABIC TEXT CLASSIFICATION WITH BOC MODEL

This section describes the developed TC system as shown in Figure 1.

The process begins by passing all the documents in the dataset through a text pre-processing phase where they are cleaned and processed. The result of this phase is a set of well-defined features. These features are then used in the modeling phase. Based on the features, i.e. words or concepts, two independent representation models are built; namely the BOW and BOC. Finally, the outputs of these phases are fed to two distinct Machine Learning (ML) based classification algorithms (classifiers). These algorithms learn from the labeled training texts to predict the class of unlabeled testing texts.

The remaining of this section elaborates on each of these phases.

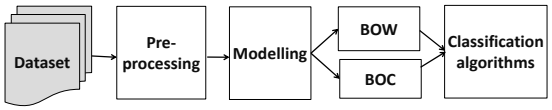

Figure 1: Developed TC system.

3.1 Text Pre-Processing

The pre-processing is the task of converting text to a well-defined set of features. In this phase all noise and irregularities, which negatively affect the classification performance, will be removed.
This phase includes the following steps:

a. All digits, punctuation marks, and non-Arabic characters are removed.
b. For normalization we follow (Kanaan et al., 2009, Mesleh, 2007) and remove diacritics and normalize some Arabic letters as follows:
   - Normalization of (ا) by replacing (أ), (إ) and (آ) at the start of words with (ا).
   - Normalization of (ي) by replacing (ى) at the end of words with (ي).
   - Normalization of (ة) by replacing (ه) at the end of words with (ة).
c. Arabic stop words and common words such as pronouns and prepositions are removed from text as they do not carry any discriminatory significance so far classification is concerned.
d. There are two slightly different approaches to stemming in Arabic language: (a) Root Extraction (RE) in which a set of prefixes, suffixes, and infixes are removed from words to extract root; and (b) Light Stemming (LS) in which only prefixes and suffixes are removed from words. Prefixes in Arabic words are groups of letters added to the beginning of the root, these letters could be definite articles, prepositions, pronouns and connectives ,such as (فال), (كال), (بال), (وال) and (ال). Suffixes in Arabic words are set of letters that are added to the end of the root, such as (يه), (ات), (ها), (ا ن), (و ن), (يه) and (ي). Infixes in Arabic words are set of letters that inserted into root to create new words; such as (ا), (ت), (و), (ن) and (ي).
   In this work, LS approach has been used as it has been found to yield better results compared

[4] http://dumps.wikimedia.org/arwiki/

to RE for Arabic TC (Kanaan et al., 2009, Mesleh, 2007).

e. Remove words that occur less than 4 times in the dataset to reduce the dimensionality of the feature vector of the representation model (Mesleh, 2007).

3.2 Text Representation

The result of pre-processing phase is a set of features that represent Arabic text. In our classification system features can be either words or concepts and, therefore two independent representation models are created, i.e., BOW and BOC. By using words as features, a BOW model is created and each word is weighted using the TFIDF weighting scheme (Salton and Buckley, 1988):

$$\mathrm{TFIDF}(w_m, d_i) = \mathrm{TF}(w_m, d_i) \cdot \mathrm{IDF}(w_m) \quad (1)$$

where $\mathrm{TF}(w_m, d_i)$ is the frequency of a word, $w_m \in W$, W is the set of all words that are considered as features.

$$\mathrm{TF}(w_m, d_i) = \frac{n(w_m, d_i)}{|d_i|} \quad (2)$$

where $n(w_m, d_i)$ is the occurrence frequency of the word w_m in document d_i, normalized by $|d_i|$ which is the length of d_i. The inverse document frequency $\mathrm{IDF}(w_m)$ is defined as:

$$\mathrm{IDF}(w_m) = \log_e \frac{|D|}{\mathrm{DF}(w_m)} \quad (3)$$

where $\mathrm{DF}(w_m)$ returns document frequency of the word w_m. It counts the number of documents in D where the word w_m appears, $|D|$ is the total number of documents in the dataset. The $\mathrm{IDF}(w_m)$ parameter has the effect of reducing the weight of those words which appear in a large portion of the dataset D.

To create the BOC model, concepts within a given text need to be identified using the Wikipedia-Miner toolkit. First, a document is processed and cleaned. Then, all possible Wikipedia concepts in the document will be identified using the topic detection functionality in Wikipedia-Miner. A majority of detected concepts are ambiguous as they may refer to multiple meanings. The disambiguation component of the toolkit is used to compute a probability estimate for each potential concept based on its relevance to other detected concepts in the document. The probability determines if a concept is relevant to the context or not and the ambiguity will be resolved. A set of candidate concepts are selected which are used to represent documents in the BOC model.

For two given documents d_1 and d_2 which belong to SNP dataset (see Section 4 for the details of this dataset); the first document belongs to 'Sport' category and describes a news article about Real Madrid football team, whereas the other document belongs to the 'Economic' category and discusses the Saudi stock market. The word Riyal (ريال) is appeared in both documents but with different intended semantic meaning. In d_1 it refers to Real Madrid (ريال مدريد) team and in d_2 it refers to Saudi Riyal (ريال سعودي) currency. By using BOW representation the difference in the meanings are lost and the Riyal (ريال) is considered the same in both documents. Furthermore, as can be seen in Table 2, the value for Riyal (ريال) document frequency in the dataset is 267 which reflects the generality of this feature. This leads to reducing the TFIDF weight of this feature and as a result the document is classified to a wrong category.

Table 2: Sample of features from SNP dataset and their Document Frequency (DF) values.

	Features				
DF	Riyal (ريال)	Madrid (مدريد)	Saudi (سعودي)	Saudi Riyal (ريال سعودي)	Real Madrid (ريال مدريد)
	267	23	581	179	9

However, by using the BOC model various meanings of the terms will be identified and mapped to their corresponding concepts, e.g., Riyal (ريال) such as Saudi Riyal (ريال سعودي), Omani Riyal (ريال عماني), Qatari Riyal (ريال قطري), Yemeni Riyal (ريال يمني) and Real Madrid (ريال مدريد), and based on the context of the document most related concepts are selected.

Table 3: Sample of weighted features in documents d_1 and d_2 with BOW representation model.

Features	d_1		d_2	
	TF	TFIDF	TF	TFIDF
Riyal (ريال)	2	0.2568	1	0.01098
Saudi (سعودي)	2	0.1892	0	0
Madrid (مدريد)	0	0	1	0.0200

Table 3 shows a sample of features with their corresponding weights in both sample documents d_1 and d_2 with the BOW model. This example demonstrates how the word Riyal (ريال) and Madrid (مدريد) have low weights in document d_2 despite their importance in the classification step. Table 4 shows that Real Madrid (ريال مدريد) weight is increased in

BOC model and reflect the importance of the concept in document d_2.

Table 4: Sample of weighted features in two documents in BOC representation model.

Features	d_1		d_2	
	TF	TFIDF	TF	TFIDF
Saudi Riyal (ريال سعودي)	2	0.4374	0	0
Real Madrid (ريال مدريد)	0	0	1	0.1359

3.3 Classification Algorithms

In the last twenty years a wide range of ML algorithms have been used for TC. The ML algorithms automatically learn from a set of pre-classified (labelled) documents.
In this work two classification algorithms have been used, namely NB and Random Forest.

a. Naive Bayes Classifier: Naive Bayes (NB) is a probabilistic classifier based on applying Baye's theorem, and is commonly used in ML applications (Mitchell, 1996). The basic idea in a NB-based classifier is to estimate the probabilities of categories for a given document by observing the joint probabilities of features and categories.

b. Random Forest Classifier: Random Forest is a commonly used classification method and it proposed by Breiman (2001). Random Forest builds a set of classification trees based on a subspace of features randomly selected to predict a category of a text instance.

4 EVALUATION

To evaluate the performance of the BOC representation model, we have assessed the accuracy of the developed classification system using two Arabic text classification datasets:

a. Arabic 1445 Dataset: This dataset has been collected by Mesleh (2007) from online Arabic newspaper archives including Al-Jazeera, Al-Nahar, Al-hayat, Al-Ahram, and Al-Dostor. The dataset contains 1445 documents that vary in length and fall into nine categories: Computer, Economics, Education, Engineering, Law, Medicine, Politics, Religion, and Sports.

b. Saudi Newspapers (SNP) Dataset: The dataset consists of 5121 Arabic documents of different lengths which belong to seven categories: Culture, Economics, General, Information Technology, Politics, Social, and Sport. It has been collected by Al-Harbi et al. (2008) and consists of articles and news stories from Saudi newspapers.

We conducted all the experiments using WEKA (Hall et al., 2009), which is a popular open source toolkit for machine learning. We first converted the textual documents into the format required by WEKA, i.e., ARFF format (Attribute-Relation File Format)[5]. We then used this data to train two separate classification algorithms namely, NB and Random Forest. This was then followed by a ten-fold cross validation to test and evaluate the performance of the classifiers, in terms of Accuracy and Macro F1 measures. These performance measures are built upon the concepts of Precision (Pr) and Recall (Re). Precision is the probability that a document predicted to be in category c_i, truly belongs to this category. Recall is the probability that a document belonging to c_i is classified into this category. When a single performance measure is desired, the harmonic mean of the precision and recall, F1, is quoted. The Accuracy (Acc) is computed by dividing the total number of documents assigned to a given category c_i by the total number of documents in the testing dataset. Let $C = (c_1, .., c_i)$ denote the set of categories in the dataset, accordingly with respect to a given category c_i:

$$\mathrm{Re}(c_i) = \frac{\text{Number of correctly assigned class labels}}{\text{Total possible correct}} = \frac{TP_i}{TP_i + FN_i} \quad (4)$$

$$\mathrm{Pr}(c_i) = \frac{\text{Number of correctly assigned class labels}}{\text{Total assigned}} = \frac{TP_i}{TP_i + FP_i} \quad (5)$$

$$F1(c_i) = \frac{2\mathrm{Pr}(c_i)\mathrm{Re}(c_i)}{\mathrm{Pr}(c_i) + \mathrm{Re}(c_i)} \quad (6)$$

$$\mathrm{Acc}(c_i) = \frac{TP_i + TN_i}{TP_i + TN_i + FP_i + FN_i} \quad (7)$$

where, the Re, Pr , F1 and Acc are computed in terms of the labels TP (True Positive), TN (True Negative), FP (False Positive) and FN (False Negative), such that:

- TP_i : refers to the cases when both the classifier and human cataloguer agree on assigning category c_i to document d;
- TN_i: refers to the cases when both the classifier and human cataloguer agree on not assigning category c_i to document d;

[5] http://www.cs.waikato.ac.nz/ml/weka/arff.html

- FP_i: refer to the cases when the classifier has mistakenly (as judged by a human cataloguer) has assigned category c_i to document d;
- FN_i: refers to the cases when the classifier has failed (as judged by a human cataloguer) to assign a correct category c_i to document d.

For automatic text classification, precision and recall values for the various classes should be combined to obtain an accurate measure of the classifier algorithms used. This is commonly done using the Macro-averaged performance, which is calculated by first computing the scores per category, i.e., Re (c_i), Pr (c_i), F1 (c_i), and then averaging these per-category scores to compute the global means.

Figure 2 compares the NB-based classifier accuracy achieved by the BOW model and BOC model as applied to both Arabic datasets. Figure 3 on the other hand compares the same for the Random Forest based classifier.

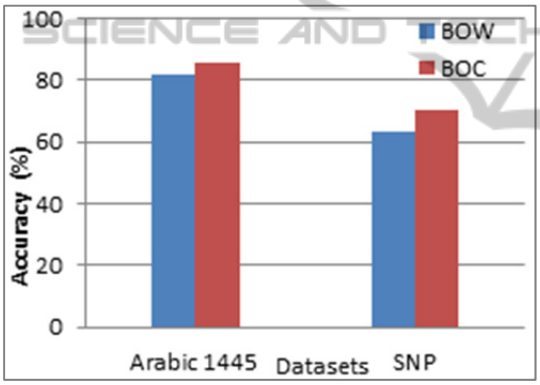

Figure 2: Ten-fold average accuracy achieved by the NB-based classifier for the BOW and BOC text representation models.

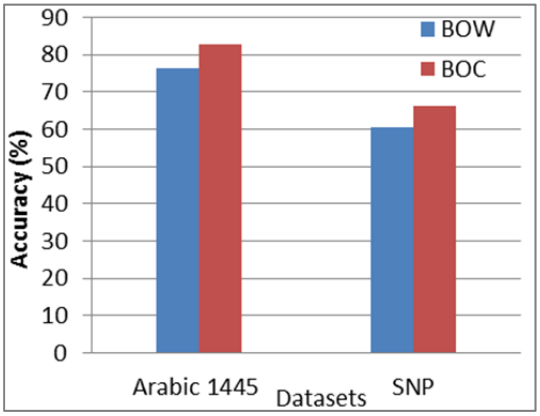

Figure 3: Ten-fold average accuracy achieved by the Random Forest -based classifier for the BOW and BOC text representation models.

Both Figures 2 and 3 indicate that the BOC representation used in our experiments for Arabic TC outperforms the BOW representation. The NB classifier seems to yield a 5% higher accuracy compared to the Random Forest classifier when applied to the Arabic 1445 dataset, and a 3% higher accuracy in the case of the SNP dataset. It has also been noted that the BOC based classifiers offer faster the execution time comapred to that of BOW's. This is due to the fact that the BOC model tends to represent a document using fewer features compared to the BOW model. Table 5 gives detials of the number of features used in each text represetnation model used in the NB classifer in our experiements and corresponding execution times of the classification algorithm in each case. In this experiment a desktop PC with a 3.33GHz Pentium processor operating under Windows 7 was used.

Table 5: Number of features and execution times for the NB-based classifier for both text representation models.

Model	Number of features	Classifier execution time (s)
BOW	12821	5.61
BOC	3463	0.96

Figure 4 shows the performance of the NB-based classifier, in terms of the Macro F1 measure, for both BOW and BOC models. The figure clearly shows that the BOC model outperforms the BOW by 10% when applied to the SNP dataset. In contrast to other classifiers that use the BOW model, our NB-based classifier with the BOC model achieved 88.99% in Macro F1 measure when applied to the Arabic 1445 dataset, as compared to 84.53% achieved by NB-based classifier reported by Kanaan et al. (2009), and 88.11% achieved by the SVM-

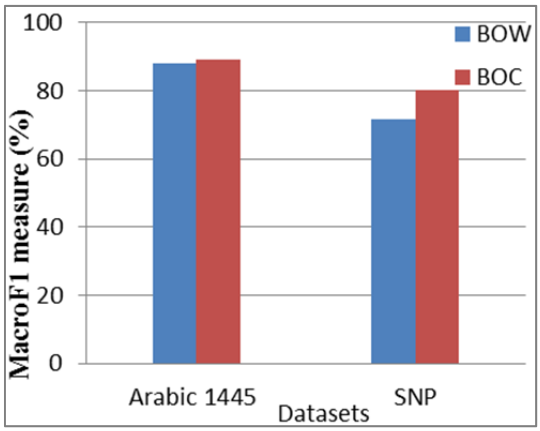

Figure 4: Performance of the NB-based classifier in terms of the Macro F1 (ten-fold averaged) for both the BOW and BOC text representation models.

based classifier reported by Mesleh (2007). For the SNP dataset, our NB-based classifier utilising the BOC model achieved 80.24% in Macro F1 measure compared to 74.0% achieved by the same type of classifier reported by Alsaleem (2011).

5 CONCLUSIONS

In this work, we demonstrated that Arabic text classification can be improved by representing textual documents as a set of concepts using the BOC model. By doing so, background knowledge can be introduced to the document representation from sources such as Wikipedia to overcome some of the limitations of the classic BOW representation. As demonstrated in our reported experimental results, the described BOC text representation model significantly improves the classification accuracy compared to the BOW model when evaluated using two Arabic TC datasets.

REFERENCES

Al-Harbi, S., Almuhareb, A., Al-Thubaity, A., Khorsheed, M. and Al-Rajeh, A. 2008. Automatic Arabic text classification.

Alahmadi, A., Joorabchi, A. and Mahdi, A. E. A new text representation scheme combining Bag-of-Words and Bag-of-Concepts approaches for automatic text classification. *GCC Conference and Exhibition (GCC), 2013 7th IEEE, 2013. IEEE*, 108-113.

Alahmadi, A., Joorabchi, A. and Mahdi, A. E. 2014. Combining Bag-of-Words and Bag-of-Concepts Representations for Arabic Text Classification. *IET Irish Signals & Systems Conference 2014.*

Alsaleem, S. 2011. Automated Arabic Text Categorization Using SVM and NB. *Int. Arab J. e-Technol.,* 2**,** 124-128.

Black, W., Elkateb, S. and Vossen, P. Introducing the Arabic wordnet project. In *Proceedings of the third International WordNet Conference* (GWC-06, 2006. Citeseer.

Breiman, L. 2001. Random forests. *Machine learning,* 45**,** 5-32.

Elberrichi, Z. and Abidi, K. 2012. Arabic Text Categorization: a Comparative Study of Different Representation Modes. *International Arab Journal of Information Technology (IAJIT),* 9.

Gabrilovich, E. and Markovitch, S. Feature generation for text categorization using world knowledge. IJCAI, 2005. 1048-1053.

Gabrilovich, E. and Markovitch, S. Overcoming the brittleness bottleneck using Wikipedia: Enhancing text categorization with encyclopedic knowledge. *AAAI*, 2006. 1301-1306.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P. and Witten, I. H. 2009. The Weka data mining software: an update. *ACM SIGKDD explorations newsletter,* 11**,** 10-18.

Hotho, A., Staab, S. and Stumme, G. 2003. Wordnet improves Text Document Clustering.

ISO. (2009). ISO-704: Terminology work—Principles and methods (3rd ed.).

Geneva, Switzerland: International Organization for Standardization.

Kanaan, G., Alshalabi, R., Ghwanmeh, S. and Alma'adeed, H. 2009. A comparison of text classification techniques applied to Arabic text. *Journal of the American society for information science and technology,* 60**,** 1836-1844.

Mesleh, A. M. D. A. 2007. Chi Square Feature Extraction Based Svms Arabic Language Text Categorization System. *Journal of Computer Science,* 3.

Milne, D. and Witten, I. H. 2013. An open-source toolkit for mining Wikipedia. *Artificial Intelligence,* 194**,** 222-239.

Mitchell, T. 1996. Machine Learning. *McCraw Hill*.

Salton, G. and Buckley, C. 1988. Term-weighting approaches in automatic text retrieval. *Information processing & management,* 24**,** 513-523.

Versteegh, K. and Versteegh, C. 1997. *The Arabic Language*, Columbia University Press.