Classified Arabic Documents Using Semi-Supervised Technique

Dr. Khalaf Khatatneh
Al-Balqa Applied University

***Abstract*—In this work, we test the performance of the Naïve Bayes classifier in the categorization of Arabic text. Arabic is rich and unique in its own way and has its own distinct features. The issues and characteristics of Arabic language are addressed in our study and the classifier was modified and regulates to fit the needs of the language. a vector or word and their frequencies method is used to represent each document. We trained our classifier using both techniques supervised and semi-supervised in an attempt to compare between them and see if the classification accuracy will improve as a result of using the technique of semi-supervised. Many various experiments were performed, and the thoroughness of the classifier was measured using recall, precision, fallout and error. The outcomes illustrates that the semi-supervised learning can significantly enhance the classification accuracy of Arabic text.**

***Keywords*—Arabic Language; Naïve Bays; Classifier; Indexing; Stop word**

I. Overview and Introduction

The classification of text is the work of sorting a set of documents into different categories from a set that was predefined. Classification of text is considered as an old domain of research but it gained more concern because the number of online documents is becoming huge and getting larger each day. Manual manipulation of this massive amount of data is extremely expensive, consuming too much time, and requires human expertise that cannot be continuously obtainable 24 hours a day, thus the need for automatic classification. Automatic text classification helps reduce the time required for classifying hundreds, even thousands, of documents every day, and will also save on the expenses and efforts of human experts.

Various algorithms for machine learning have been used for the process of text categorization: support vector machines, *k* means the closest neighbor, naïve Bayes, and neural networks considered as some of the most common ones, most of which were found to work quite well through the area of text classification. In our work, the naïve Bayes was chosen as a classifier. An applying Bayes' theorem (from Bayesian statistics) was used as a straightforward probabilistic classifier for Naïve Bayes along with using strong naïve independence assumptions; it supposes that the existence/absence of a specific word in/from a class (category) is separated from the existence/absence of another word. Despite the fact that naïve Bayes is simple and makes over-simplified assumptions, it has assured to completely work in a good manner in many complex real-world positioning, and has been known to produce very good results, with high classification accuracy [9][14]. These reasons contributed to our decision to choose naïve Bayes to be our classifier.

Researches in the text categorization area were mainly restricted to English text. Many studies also contain various continental languages. For example, German, French, and Spanish as well as languages of Asian countries such as Japanese and chine's. Researches that address text that written using Arabic language is rare in literature. [2] Attempted to attain a better understanding of Arabic text classification by evaluating the rendering of two widespread algorithms of classification (SVM and C5.0) that used in the process of classifying text that written using Arabic. Another contribution is presented in the works of [10], in which highest entropy was the method that used for classifying Arabic documents. The *k*-nearest neighbor algorithm was used in [4] for Arabic text classification in an attempt to assess the performance of this algorithm in the process of classifying text written using Arabic. Finally, Rehab M. Duwairi contributed to the research in Arabic text classification in two of her papers; [8] and [9] where she compares the accuracy of three classifiers (distance-based, Naïve Bayes, and *knn*) when used for categorizing Arabic text.

In the current research we test the precision of the Naïve Bayes classifier in categorizing Arabic text using both supervised and semi-supervised techniques in an attempt to compare between them and see if the classification accuracy will increase as a result of using the semi-supervised technique. In the supervised approach the classifier first trains using a collection of labeled documents and is then given a collection documents that are unlabeled in order to automatically classify based on the information it has gained in the training process. The semi-supervised approach, however, takes it one step further; not only does the classifier train on labeled documents, but also uses unlabeled documents for training.

Training data set was collected from many online forums, magazines, and newspapers. We had a total of 1890 documents that varied in length and writing style. These documents fall into 9 different categories with a various documents' number for every category. The data set was divided into three groups: the labeled training documents, the unlabeled data, and the data set used for testing. The preprocessed documents were provided by removing all stopwords, symbols, and digits, and light stemming was used by removing some of the prefixes and suffixes from the keywords.

The dataset was divided into 70% labeled data, 20% unlabeled data, and 10% test data. The results of this experiment were measured in recall, precision, fallout and error rate. The recall value that represents the classification accuracy of the supervised learning method was 77% and the one for semi-supervised learning method was 87%. It was also proven that the semi-supervised learning method's accuracy could not be further improved if it was given an extra number of documents to classify and learn from.

The remaining of the work is arranged as: second part describes the unique features of the Arabic language and the main issues that were taken into account when the classifier was built. In third part we explain about the preprocessing we performed on our documents. Fourth part includes an explanation about the Naïve Bayes classifier, the supervised and semi-supervised approaches, the implementation of the classifier, and the final results. Finally, fifth part contains the entire conclusion for this work.

II. Characteristics of Arabic

The central HYPERLINK "http://en.wikipedia.org/wiki/Central_Semitic_language"SemiticHYPERLINK "http://en.wikipedia.org/wiki/Central_Semitic_language" language is Arabic; therefore it has some relations with other Semitic languages. For example, languages of Hebrew and Neo-Aramaic. Arabic is the language that has additional speakers than any other Semitic language. It is used by huge number of users that exceeds 280 million people, and is considered as the official language of 22 countries [5].

The alphabet of Arabic consists of 28 characters:

ا ب ت ث ج ح خ د ذ ر ز س ش ص ض ط ظ ع غ ف ق ك ل م
و ي ن هـ

Besides to the Arabic *hamza* (ء), which is often considered to be a letter. Three of the letters are vowels (ا و ي), while the rest of the letters are consonants. Arabic text is written from right to left. The letters of this language take different forms and shapes depending on two main things: first, their position within the word (first, middle, or end), and second, whether the letter is connectable to its next neighbors.

Arabic is a highly inflected language; In addition, a verb in its root pattern is augmented with prefixes, infixes, and suffixes to reflect the time during which the event occurred, whether the verb is plural or singular (plural is divided into *two* and *three or more*) as well as the gender of the participants in the verb.

Arabic used diacritics. Diacritics are short vowels that are written above or below a letter to indicate the pronunciation on the letter. There are four main diacritics: *fat-ha*, *damma*, *kasra*, and *shadda*, in addition to *double fat-ha* (called *tanween fateh*), *double damma* (called *tanween damm*), *double kasra* (called *tanween kaser*), and *sukun*.

Arabic may vary in meaning depending on the diacritics. Some words in Arabic vary in meaning if the diacritics change. In this case, if diacritics are not added to clarify the meaning of the word, it is considered to be ambiguous. For more illustration on ambiguous words, we site the following examples:

- The word (شِفاه) means lips. Note that there is a kasra under the first letter (ش or sheen). When this kasra is replaced with a fat-ha placed above the letter, the word becomes (شَفاه) which means cured.
- The word (نَهْرْ) means *river*. Note that there is *sukun* above both letters (هـ or *ha*) and (ر or *ra*). When both *sukuns* are replaced with a couple of *fat-has*, the word becomes (نَهَرَ) which means *scolded*.

On the other hand, many of the words do not change in meaning due to change in diacritics. The change in diacritics in their case produces words that have no meaning, and so the meaning of the word is clear even if diacritics are suppressed.

Diacritics have a big task in the meaning of the word. Unfortunately, the majority of Arabic text is written without diacritics. This is a big issue in the text classification problem, and leads to one of the complications of Arabic text categorization in contrast to the English language. The diacritics have a big task in the meaning of the word, however, most of the time they are ignored and omitted causing many words to lose their meaning or to be confused with other words (it is better to find a method to deal carefully with diacritics in "Preprocessing phase"). As a result, misclassifications are bound to happen, causing a decrease in the classification accuracy.

III. Document Preprocessing

We pre-processed the documents in two ways: filtering and stemming. We filtered out any word that occurred less than 5 times in a document. Multiple experiments were held and we concluded that the removal of the words that appear less than 5 times improves the performance of the classifier.

Since Arabic is a highly inflected language, we, also, performed light stemming. As mentioned earlier, often an Arabic verb in its root form is augmented with prefixes, infixes, and suffixes. Fortunately, all Arabic words can be mapped to their root types. Arabic words can have three-, four-, five-, or six-letter roots. More than 80% of Arabic words have three-letter roots [8]. The process of root extracting from a word is called root stemming. Stemming in general includes removing any added prefixes and suffixes to the word, and it is much needed in the text classification problem for the purpose of reducing the dimensionality of the feature vector. According to [1], there are two kinds of stemming:

1) Root stemming: a technique that attempts to reduce the word to its original root.

2) Light stemming: a technique that attempts to remove only some of the prefixes and/or suffixes. It does not attempt to remove any infixes or reduce the word to its root form [3].

TABLE I. The Prefixes and Suffixes that were Eliminated by Light Stemming

Prefixes	وكال – كال – فال – وبال – بال – لل – وال – ال
Suffixes	ها

In this work, we chose to do some light stemming on the documents rather than root stemming. Our light stemming started by removing all the diacritics from our documents (it is more accurate when using diacritics), and then we removed the prefixes and suffixes, shown in Table 1 from all the words.

IV. Supervised and Semi-Supervised Naïve Bayesian Classification

One of the probabilistic classifier that considered simple is Naive Bayesian classifier [7] [6] that uses theorem of Bayes for conditional probabilities. It is called naïve; because it assumes that all values of the attribute are independent from each other given a class value (i.e. it supposes that the presence or absence of a certain feature of a class is unrelated to the presence or absence of any other feature). Despite this naïve assumption, Naïve Bayesian has been successfully used as a text classifier [13][12][11][8].

To classify a new document, the method calculates the probability of each class value, given the document's words. The maximum probability of the class is then taken as the predicted class of the document. The training set is used to estimate all needed probabilities.

Given a document that contains the words $w_1, w_2, \ldots, w_n$, a value of a class has the probability , C, is computed as

$$P(C / w_1, w_2, \ldots, w_n) = \frac{P(w_1, w_2, \ldots, w_n / C)P(C)}{P(w_1, w_2, \ldots, w_n)} \quad (1)$$

Where:

P(C) is considered as the probability of class *C*.

$P(w_1, w_2, \ldots, w_n)$ is the probability that words $w_1, w_2, \ldots, w_n$ occur in a document irrespective of their position in the document.

$P(w_1, w_2, \ldots, w_n / C)$ is the words $w_1, w_2, \ldots, w_n$ will appear in a document of class C.

Since, given a document, the probability $P(a_1,a_2,\ldots,a_n)$ is the same regardless of the class, therefore, formula 1 can be simplified as follows:

$$P(C / w_1, w_2, \ldots, w_n) = P(w_1, w_2, \ldots, w_n / C)P(C) \quad (2)$$

The approach got its name because it naively assumes that values of the attribute are conditionally distinct stated the value of the class. Therefore, it assumes that

$$P(w_1, w_2, \ldots, w_n / C) = \prod_i P(w_i / C) \quad (3)$$

The documents are modeled as groups of words where the i-th word of a certain document has the probability that happens from class C in a document is written as P (wi|C). It is assumed that the position of the word within the document is not relevant.

A. Semi-Supervised Text Classification

Supervised learning uses a training set that consists of manually classified documents. Naïve Bayesian uses this training set to estimate all required probabilities. Therefore, the larger this set, the more accurate the estimations are. However, preparing a large training set is a tedious task that requires effort and time.

Semi-supervised classification [14] attempts to make use of unlabelled (unclassified) documents to increase the classification accuracy of a classifier; Initially, just like a supervised approach, the classifier is trained using a set of classified documents. The classifier is, then, given a set of unlabeled documents to classify. The newly classified documents are then added to the pool of training documents and the new bigger set is then used to re-estimate all needed probabilities. So actually, the algorithm is learning partially from unlabeled data.

The two main steps in semi-supervised learning are called EM [14]:

(E-step): utilize the naïve Bayes classifier to approximate the classification for each unlabeled document.

(M-step): the classifier is re-estimated given the new labeled documents.

V. Experiments and Results

Our data set was gathered from online forums, magazines and newspapers. We used a total of 1893 documents that vary in length and writing style. The documents fall into 9 different classes: Economics, Computer Science, Education, Engineering, Politics, Law, Religion, and Sports, with a different number of documents for each class. The whole documents for each classification are shown in Table 2**Error! Reference source not found.**

TABLE II. The # of Documents in Each Class

Category	# of Documents
Computer	120
Economics	270
Education	118
Engineer	165
Law	147
Medicine	283
Politics	232
Religion	277
Sport	282
Total	**1893**

The number of fold cross validation that used in our experiments is ten. This means that the each experiment was repeated 10 times, using a different subset of 10% of a test set of as the training data, each time. In each fold, the training data, which comprise of 90% of the original data set, was partitioned into 70% labeled documents, used to train the classifier in a supervised way, and 20% unlabeled documents used to, further, train the classifier in a semi-supervised way.

At each fold, the classifier was trained in a supervised way

using only 70% of the original training data, then the accuracy was measured using the test data. This accuracy is reported as the result of supervised learning. The classifier was then further trained, using the unlabeled documents, in a semi-supervised way. The same test data was used to measure the classification accuracy. This accuracy is reported as the accuracy of semi-supervised training. This process was repeated 10 times, using a different test set of 10% data each time. Table 4 shows the average 10-fold classification accuracy for each category of documents. Semi-supervised learning was performed as batch learning; in the sense that, all unlabeled documents were labeled (classified) first, and then the probabilities were re-calculated. Also, the vocabulary list was updated to include the new words that appeared in the unlabelled documents (as "features extraction" process).

We can determine the accuracy of the classifier by expressing terms of recall, precision, fallout, and error percentage . To enlarge elaboration on the formulae of the four terms consider a binary classification matter (i.e., there are only one category and n documents that require to be classified), so a given document either belongs to this category (i.e., positive example) or does not belong to that category (i.e., negative example). presume that the classification is carried out by two classifiers: the first is a human and the second is a computer program. Then *recall (Re), precision (Pr), both of fallout,* and *error rate* are calculated as

$$Re = \frac{a}{(a+c)}$$

$$Pr = \frac{a}{(a+b)}$$

$$Fallout = \frac{b}{(b+d)}$$

$$Error\ rate = \frac{(b+c)}{(a+b+c+d)}$$

Where a = number of documents that both the human and the computer classify as positive examples, b = number of documents that the human classifies as negative examples but the computer classifies as positive examples, c = number of documents that the human classifies as positive examples but the computer classifies as negative examples, d = number of documents that both the human and the computer classify as negative documents, and $a + b + c + d = n$ (all test documents) [8].

Table 3 shows the result of comparing supervised learning and semi-supervised learning in terms of four accuracy measures: recall, precision, fallout, and classification error. It is obvious from the table that semi-supervised learning improved the results in terms of the four accuracy measures. The average recall of supervised learning is 76.33%. It rose to 84.67% using semi-supervised learning. Similarly, precision rose from 78.87% using supervised learning to 85.37% using semi-supervised learning. The fallout and error, also, fall down from 2.58% and 4.73% to 1.81% and 3.09%, respectively.

TABLE III. THE CLASSIFICATION ACCURACY FOR EACH CATEGORY (CLASS) OF DOCUMENTS USING SUPERVISED AND SEMI-SUPERVISED METHODS

	Supervised				**Semi-Supervised**			
	Recall	Prec.	Fall	Error	Recall	Prec.	Fall	Error
Computer	88.00%	94.20%	0.40%	1.10%	88.00%	92.40%	0.50%	1.20%
Economics	79.00%	66.50%	6.10%	7.20%	80.00%	77.80%	3.50%	5.70%
Education	58.00%	83.60%	0.70%	3.20%	62.00%	83.50%	0.80%	3.00%
Engineer	83.00%	75.60%	0.30%	4.10%	89.00%	92.40%	0.80%	1.70%
Law	72.00%	50.80%	6.00%	7.70%	75.00%	67.10%	3.10%	4.80%
Medicine	81.00%	93.30%	1.10%	2.00%	93.00%	98.10%	0.30%	1.40%
Politics	76.00%	74.90%	3.70%	6.30%	85.00%	79.70%	3.10%	4.70%
Religion	87.00%	80.50%	3.80%	5.10%	92.00%	84.50%	3.00%	3.80%
Sport	63.00%	90.40%	1.10%	5.90%	98.00%	92.80%	1.20%	1.50%
Average	**76.33%**	**78.87%**	**2.58%**	**4.73%**	**84.67%**	**85.37%**	**1.81%**	**3.09%**

At this point, one issue merits further investigation. Will the classifier give better performance if it was fed more unlabeled documents to classify and then learn from (i.e. will semi-supervised learning continue to improve the results)?

To answer this question, we collected (downloaded), yet, another set of documents. This new set consisted of 90 documents, 10 documents of each category. We trained the classifier (that we got of the semi-supervised phase) in a semi-supervised way, using the new 90 documents. We compared the results of the two experiments to see if there is any improvement on the classification accuracy of the algorithm; the results showed no improvement and the classification accuracy of the classifier remained the same as the one in the original experiment. This experiment does not prove that no further improvement is possible using semi-supervised learning, but at least, it shows that further improvement becomes more difficult to achieve as the error rate becomes smaller.

VI. A ROUGH SET-BASED APPROACH

Rough set methods can be used and applied here to improve the classification accuracy by feature selection. These methods based on mathematical and statistical calculations drive the algorithm to eliminate some of attributes. [15][16]

VII. FEATURES EXTRACTION BY PFC

Systematic features extraction is a main process of documents classification. Hence, taking a care of this phase does not lost the time, it is a valuable investigation to choose a clever method and fast to extract feature from the given documents. Using PFC (principle-feature classification) to extract the feature using a sequential method and pruning the used data may give the algorithm more efficiency and accuracy. [17]

VIII. CONCLUSION

This work demonstrates that the learning of semi-supervised can develope the accuracy of classification for Arabic documents, but this improvement becomes more difficult as the error rate becomes smaller. We used the Naïve Bayesian algorithm as to train the classifier. As the Arabic language is a highly inflected language, we performed light

stemming on the documents. Semi-supervised learning gave better results than supervised learning only when we used batch learning and allowed the list of vocabulary to be dynamic. It turned out that adding the new words that appeared in the new documents to the list of vocabulary during training was essential to improve the classification accuracy.

REFERENCES

[1] Al-Ameed H., Al-Ketbi S., Al-Kaabi A., Al-Shebli K., Al-Shamsi N., Al-Nuaimi N. and Al-Muhairi S., (2005), "*Arabic Light Stemmer: A new Enhanced Approach"*, The Second International Conference on Innovations in Information Technology (IIT'05).

[2] Al-Harbi S., Almuhareb A., Al-Thubaity A., Khorsheed M. S., Al-Rajeh A., (2008), "Automatic Arabic Text Classification", In *Proceedings of The 9th International Conference on the Statistical Analysis of Textual Data*, Lyon-France.

[3] Al-Kharashi I. And Al-Sughaiyer I. , (2004), "*Performance Evaluation of an Arabic Rule-Based Stemmer"*, The 7th National Conference on Infromation Technology and Computers, King Abdulaziz University, Saudi Arabia.

[4] Al-Shalabi R., Kanaan G., and Gharaibeh M. H., (2006), "*Arabic Text Categorization Using kNN Algorithm"*, Proceedings of The 4th International Multiconference on Computer Science and Information Technology, Vol. 4, Amman, Jordan.

[5] Britannica, 2011. Retrieved in April, 2011 from http://www.britannica.com/EBchecked/topic/31677/Arabic-language

[6] Domgos P, Pazzani M. (1996). "Beyond Independence: conditions for the optimality of the Simple Bayesin Classifier". The 13th International Conference on Machine Learning.

[7] Duda R. , Hart P. (1973). "*Pattern Classification and Scene Analysis"*. John Wiely and Sons.

[8] Duwairi R. M., (2006), "*Machine Learning for Arabic Text Categorization"*, Journal of the American Society for Information Science and Technology, Vol. 57, No. 8, p. 1005-1010.

[9] Duwairi R. M., (2007), "*Arabic Text Categorization"*, The international Arab Journal of Information Technology, Vol. 4, No. 2.

[10] El-Halees A. M., (2007), "*Arabic Text Classification Using Maximum Entropy"*, The Islamic University of Gaza, Journal of Series of Natural Studies and Engineering Vol. 15, No. 1, pp 157-167.

[11] Manning C., R. P. (2008). "*Introduction to Information Retrieval"*. Cambridge University Press.

[12] McCallum A. and Nigam K. (1998). "A comparison of event models for naive Bayes text classification". In AAAI workshop on learning for text categorization.

[13] Mitchell T. M., (1997), "*Machine Learning"*, McGraw-Hill.

[14] Nigam K., McCallum A., Mitchell T. M., 2006, "*Semi-supervised Text Classification Using EM"*, In Semi-supervised Learning, O. Chapelle, A. Zien, and B. Scholkopf (Eds.), MIT Press.

[15] Alexios Chouchoulas1, Qiang Shen1., 1999, "*A Rough Set-Based Approach to Text Classification*".

[16] Libiao Zhang⁎, Yuefeng Li†, Chao Sun‡, Wanvimol Nadee§., 2013, "*A Rough Set-Based Approach to Text Classification*".

[17] Donald W. Tufts and Qi Li., 1997, "*PRINCIPAL-FEATURE CLASSIFICATION*".