Automatic Arabic Document Categorization Based on the Naïve Bayes Algorithm

Mohamed EL KOURDI **Amine BENSAID**♠ **Tajje-eddine RACHIDI**

School of Science & Engineering
Alakhawayn University
P.O. Box 104, Ifrane 53000, Morocco
[M.Elkourdi, A.Bensaid, T.Rachidi]@alakhawayn.ma
♠Corresponding Author

Abstract

This paper deals with automatic classification of Arabic web documents. Such a classification is very useful for affording directory search functionality, which has been used by many web portals and search engines to cope with an ever-increasing number of documents on the web. In this paper, Naive Bayes (NB) which is a statistical machine learning algorithm, is used to classify non-vocalized Arabic web documents (after their words have been transformed to the corresponding canonical form, i.e., roots) to one of five pre-defined categories. Cross validation experiments are used to evaluate the NB categorizer. The data set used during these experiments consists of 300 web documents per category. The results of cross validation in the leave-one-out experiment show that, using 2,000 terms/roots, the categorization accuracy varies from one category to another with an average accuracy over all categories of 68.78 %. Furthermore, the best categorization performance by category during cross validation experiments goes up to 92.8%. Further tests carried out on a manually collected evaluation set which consists of 10 documents from each of the 5 categories, show that the overall classification accuracy achieved over all categories is 62%, and that the best result by category reaches 90%.

Keywords: Naïve Bayes, Arabic document categorization, cross validation, TF-IDF.

1 Introduction

With the explosive growth of text documents on the web, relevant information retrieval has become a crucial task to satisfy the needs of different end users. To this end, automatic text categorization has emerged as a way to cope with such a problem. Automatic text (or document) categorization attempts to replace and save human effort required in performing manual categorization. It consists of assigning and labeling documents using a set of pre-defined categories based on document contents. As such, one of the primary objectives of automatic text categorization has been the enhancement and the support of information retrieval tasks to tackle problems, such as information filtering and routing, clustering of related documents, and the classification of documents into pre-specified subject themes. Automatic text categorization has been used in search engines, digital library systems, and document management systems (Yang, 1999). Such applications have included electronic email filtering, newsgroups classification, and survey data grouping. Barq for instance uses automatic categorization to provide similar documents feature (Rachidi et al., 2003). In this paper, NB which is a statistical machine learning algorithm is used to learn to classify non-vocalized[1] Arabic web text documents.

This paper is organized as follows. Section 2, briefly describe related works in the area of automatic text categorization. Section 3 describes the preprocessing undergone by documents for the purpose of categorization; it describes in particular the preprocessing specific to the Arabic language. In section 4 Naïve Bayes (NB), the learning algorithm used in this paper for document categorization is presented. Section 5 outlines the experimental setting, as well as the experiments carried out to evaluate the performance of the NB classifier. It also gives the numerical results with their analysis and interpretation. Section 6 summarizes the work and suggests some ideas for future works.

2 Related Works

Many machine learning algorithms have been applied for many years to text categorization. They

[1] Most modern Arabic writing (web, novels, articles) are written without vowels.

include decision tree learning and Bayesian learning, nearest neighbor learning, and artificial neural networks, early such works may be found in (Lewis and Ringnette, 1994), (Creecy and Masand, 1992) and (Wiene and Pedersen, 1995), respectively.

The bulk of the text categorization work has been devoted to cope with automatic categorization of English and Latin character documents. For example, (Fang et al., 2001) discusses the evaluation of two different text categorization strategies with several variations of their feature spaces. A good study comparing document categorization algorithms can be found in (Yang and Liu, 1999). More recently, (Sebastiani, 2002) has performed a good survey of document categorization; recent works can also be found in (Joachims, 2002), (Crammer and Singer, 2003), and (Lewis et al., 2004).

Concerning Arabic, one automatic categorizer has been reported to have been put under operational use to classify Arabic documents; it is referred to as "Sakhr's categorizer" (Sakhr, 2004). Unfortunately, there is no technical documentation or specification concerning this Arabic categorizer. Sakhr's marketing literature claims that this categorizer is based on Arabic morphology and some research that has been carried out on natural language processing.

The present work evaluates the performance on Arabic documents of the Naïve Bayes algorithm (NB) - one of the simplest algorithms applied to English document categorization (Mitchell, 1997). The aim of this work is to gain some insight as to whether Arabic document categorization (using NB) is sensitive to the root extraction algorithm used or to different data sets. This work is a continuation of that initiated in (Yahyaoui, 2001), which reports an overall NB classification correctness of 75.6%, in cross validation experiments, on a data set that consists of 100 documents for each of 12 categories (the data set is collected from different Arabic portals). A 50% overall classification accuracy is also reported when testing with a separately collected evaluation set (3 documents for each of the 12 categories). The present work expands the work in (Yahyaoui, 2001) by experimenting with the use of a better root extraction algorithm (El Kourdi, 2004) for document preprocessing, and using a different data set, collected from the largest Arabic site on the web: aljazeera.net.

3 Preprocessing of document

Prior to applying document categorization techniques to an Arabic document, the latter is typically preprocessed: it is parsed, in order to remove stopwords (these are conjunction and disjunction words etc.). In addition, at this stage in this work, vowels are stripped from the full text representation when the document is (fully or partially) voweled/vocalized. Then roots are extracted for words in the document.

In Arabic, however, the use of stems will not yield satisfactory categorization. This is mainly due to the fact that Arabic is a non-concatenative language (Al-Shalabi and Evens, 1998), and that the stem/infix obtained by suppression of infix and prefix add-ons is not the same for words derived from the same origin called the root. The infix form (or stem) needs further to be processed in order to obtain the root. This processing is not straightforward: it necessitates expert knowledge in Arabic language word morphology (Al-Shalabi and Evens, 1998). As an example, two close roots (i.e., roots made of the same letters), but semantically different, can yield the same infix form thus creating ambiguity.

The root extraction process is concerned with the transformation of all Arabic word derivatives to their single common root or canonical form. This process is very useful in terms of reducing and compressing the indexing structure, and in taking advantage of the semantic/conceptual relationships between the different forms of the same root. In this work, we use the Arabic root extraction technique in (El Kourdi, 2004). It compares favorably to other stemming or root extraction algorithms (Yates and Neto, 1999; Al-Shalabi and Evens, 1998; and Houmame, 1999), with a performance of over 97% for extracting the correct root in web documents, and it addresses the challenge of the Arabic broken plural and hollow verbs. In the remainder of this paper, we will use the term "root" and "term" interchangeably to refer to canonical forms obtained through this root extraction process.

4 NB for document categorization

4.1 The classifier module

The classifier module is considered to be the core component of the document categorizer. It is responsible for classifying given Arabic documents to their target class. This is performed using the Naive Bayes (NB) algorithm. The NB classifier computes a posteriori probabilities of classes, using estimates obtained from a training set of labeled documents. When an unlabeled document is presented, the a posteriori probability is computed for each class using (1) in Figure 1; and the unlabeled document is then assigned to the class with the largest a posteriori probability.

A posteriori probability computation

Let D be a document represented as a set of finite terms $D=\{w_1, w_2,..., w_3\}$.

Let C be the number of target classes.

Let $docs_i$ *be the number of documents in category* $C,_i$ *and |Examples| be the number of documents in the training set of labeled documents.*

Let n be the total number of distinct stems in C_i

Let N_k *be the number of times* w_k *occurs in* C_i

Then the a posteriori probability as given by Bayes theorem is:

$$P(C_i|D)=[P(C_i)*P(D|\ C_i)]/P(D).\ i=1,2,...C \qquad \textbf{(1)}$$

When comparing a posteriori probabilities for the same document D, P(D) is the same for all categories and will not affect the comparison.

The other quantities in (1) are estimated from the training set using NB learning (see Figure 2).

The assigned class AC(D) to document D is the class with largest a posteriori probability given by (1):

$$AC(D)=argmax_{Ci}\ \{\ P(C_i|D).\ i=1,2,...C\}$$

Figure 1. A posteriori probability reduction.

4.2 The learning module

The main task of the learning module is to learn from a set of labeled documents with predefined categories in order to allow the categorizer to classify the newly encountered documents *D* and to assign them to each of the predefined target categories C_i. This module is based on the NB learning algorithm given in Figure 2. The learning module is one way of estimating the needed quantities in (1) by learning from a training set of documents.

NB learning algorithm

Let D be a document represented as a set of finite terms/roots $D=\{w_1, w_2,..., w_n\}$.

Let $docs_i$ *be the number of documents in category* C_i *, and |Examples| be the number of documents in the training set of labeled documents.*

Step 1: collect the vocabulary, which is defined as the set of distinct words in the whole training set

Step2: For each category C_i *do the following*

Compute $P(C_j) = |\ docs_j\ |/|Examples|$ ***(2)***

where $docs_j$ *is the number of training documents for the category is* C_j.

*F*or each root w_k in Vocabular*y*

Compute $P(w_k/C_j)= (N_{k,j} +1)/(\ n_j +|\ Text_j\ |)$ ***(3)***

where $N_{k,j}$ *is the number of times* w_k *occurs in* C_j, n_j *is the total number of distinct terms in all training documents labeled* C_j, *and* $Text_j$ *is a single documents generated by concatenating all the training documents for category* C_j .

Equation (2) and (3) make use of the following two assumptions:

1) Assuming that the order of the words in a document does not affect the classification of the document:

$$P(D|C_j)=P(\{w_1, w_2,..., w_n\}|C_j) \qquad \textbf{(4)}$$

2) Assuming that the occurrence of each word is independent of the occurrence of other words in the document then:

$$P(w_1,...,w_n|C_j)=P(w_1|C_j)*P(w_2|C_j)*...*P(w_n|C_j) \quad \textbf{(5)}$$

Figure 2. The Naïve Bayes (supervised) learning algorithm for document categorization

The m-estimate method (with m equal to the size of word vocabulary) (Cestink, 1990) is used to compute the probability terms and handle zero count probabilities (smoothing). Equation (3) gives an estimate for $P(w_k/C_j)$.

Various assumptions are needed in order to simplify Equation (1), whose computations are otherwise expensive. These assumptions are applied in Figure 2 to obtain the needed quantities for the class-conditional probabilities (Equations (4) and (5)). These assumptions are:

1. The probability of encountering a specific word within a document is the same regardless the word position. In other words, $P(w_i=w|C_j)= P(w_m= w|C_j)$ for every i, j, and m where i and m are different possible positions of the same word within the document. This assumption allows representing a document as a bag of word (Equation (4) in Figure 2).

2. The probability of occurrence of a word is independent of the occurrence of other words in the same document. This is reflected in Equation (5): $P(w_1,...,w_n|C_j)=P(w_1|C_j)*P(w_2|C_j)*...*P(w_n|C_j)$. It is in fact a naïve assumption, but it significantly reduces computation costs, since the number of probabilities that should be computed is decreased. Even though this assumption does not hold in reality, NB performs surprisingly well for text classification (Mitchell, 1997).

5 Experiments and results

For classification problems, it is customary to measure a classifier's performance in terms of classification error rate. A data set of documents is used with known category/class label $L(D_k)$ for each document D_k. The set is split into two subsets: a training set and a testing set. The trained classifier is used to assign a class $AC(D_k)$ using Equation (3) to each document (D_k) in the test set, as if its true class label were not known. If $AC(D_k)$ matches $L(D_k)$, the classification is considered correct; otherwise, it is counted as an error:

$$\text{Error}_{ik}=\left\{\begin{array}{ll} 1 & \text{iff } L(D_k) = C_i, \text{ and } AC(D_k) \neq C_i \\ 0 & \text{otherwise} \end{array}\right\} \qquad \textbf{(6)}$$

For a given class, the error rate is computed as the ratio of the number of errors made on the whole test set of unlabeled documents (X^u) to the cardinality $|X^u|$ of this set. For a given class C_i, the error rate is computed as:

$$\text{ClassError}_i = \sum_{k=1}^{|X^u|} \text{Error}_{ik} \,/\, |X^u| \qquad \textbf{(7)}$$

In order to measure the performance of the NB algorithm on Arabic document classification, we conducted several experiments: we performed cross validation using the original space (using all the words in the documents), cross validation experiments based on feature selection (using a subset of terms/roots only), and experiments based on an independently constructed evaluation set. The following paragraphs describe the data set used, and the experiments.

5.1 The data set

We have collected 300 web documents for each of five categories from the website www.aljazeera.net, which is the website of Aljazeera (the Qatari television news channel in Arabic). This site contains over seven million (7,000,000) documents corresponding to the programs broadcast on the television channel; it is arguably the most visited Arabic web site. Aljazeera.net presents documents in (manually constructed) categories. The five (5) categories used for this work are: sports, business, culture and art, science, and health.

5.2 Cross validation

In cross validation, a fixed number of documents is reserved for testing (as if they were unlabeled documents) and the remainder are used for training (as labeled documents). Several such partitions of the data set are constructed, by making random splits of the data set. NB's performance is evaluated several times, using the different random partitions. Then the error statistics are aggregated. The steps of the cross validation experiments are delineated in Figure 3 next:

Cross validation steps

Let X be the entire data seto f N=1500 documents

c =5 is the number of different categories

$E_{r,i}$ will store the error rate for category i during trial r.

1) Fix the size s of the training set for (s=N/3, N/2, 2N/3, or N-1) to perform 1/3-2/3, 50/50, 2/3-1/3 or leave-one-out cross validation.

2) Set the number of trials T. If s=N-1, fix the number of trials T=N; else, T=40.

3) For trial r=1 to T

3.1 Select randomly s documents from X as labeled documents into training set X_r^l.

3.2 Store the remaining documents (X- X_r^l) as unlabeled documents into X_r^u (as if they were unlabeled).

3.3 Train NB using X_r^l. (Compute Equation (2) and Equation (4))

3.4 Use trained NB to compute the class of each element in X_r^u using Equation (4)

```
    3.5 Compute error rate $E_{r,i}$ , on $X^{u}_{r}$ for each category (i=1,2...,c) using Equation (7):
    $E_{r,i} = \sum_{k=1}^{|X^u|} Error_{ik} / |X^u|$   i=1,2,…,c
    Next r (return to step 3).
    4.1 Compute the average error rate for each class over all trials:
    $AvgError_{i,s}. = \sum_{r=1}^{T} E_{r,i} / T$   i=1,2,…,c
    4.2 Compute the maximum error rate for each class over all trials:
    $MaxError_{i,s} = Max_{r=1,2...,T} \{E_{r,i}\}$   i=1,2,…,c
    4.3 Get the minimum error rate for each class over all trials:
    $MinError_{i,s} = Min_{r=1,2...,T} \{E_{r,i}\}$   i=1,2,…,c.
    Next s (return to step 1)
```

Figure 3. Cross validation experiments.

5.2.1. Experiments without feature extraction

In these experiments, each document in data set X is represented by all word roots in the document. The cross validation experiments described in Figure 3, is conducted. Table 1 reports the error rates obtained over all categories during the cross validation experiments. The smallest error rate is obtained in the leave-one-out experiment (as illustrated in Table 1). Table 2, Table 3, Table 4, and Table 5 represent, respectively, the confusion matrices of the cross validation experiments. The percentages reported in an entry of a confusion matrix correspond to the percentage of documents that are known to actually belong to the category given by the row header of the matrix, but that are assigned by NB to the category given by the column header.

		Cross-validation Experiments			
		1/3-2/3	1/2-1/2	2/3-1/3	Leave-one-out
Error Rate	Avg	67%	55%	46%	32.1%
	Max	69.9%	56.5%	49%	100%
	Min	62%	48.1%	42%	0%

Table 1. The error rates of NB over all categories in cross validation experiments (with feature extraction)

Category	Health	Business	Culture	Science	Sport
Health	22%	27%	3%	8%	40%
Business	7%	39%	10%	18%	26%
Culture	13%	18%	27%	7%	35%
Science	14%	15%	8%	30%	33%
Sport	16%	12%	17%	8%	47%

Table 2. Confusion Matrix results for cross validation, with no feature extraction (1/3-2/3).

Category	health	Business	Culture	Science	Sport
Health	32%	22.5%	3.2%	8%	34.3%
Business	8.2%	50%	10.7%	13.3%	17.8%
Culture	8%	20%	39%	3%	30%
Science	16%	9.8%	7.2%	46%	21%
Sport	12%	8%	16%	4%	60%

Table 3. Confusion Matrix results for cross validation, with no feature extraction (1/2-1/2).

Category	Health	Business	Culture	Science	Sport
Health	46%	12%	6%	8%	28%
Business	4.8%	63%	7%	9.2%	16%
Culture	7.1%	16.8%	42%	6.1%	28%
Science	8.1%	10.8%	9.1%	46%	26%
Sport	7.2%	5%	6.8%	5%	76%

Table 4. Confusion Matrix results for cross validation, with no feature extraction (2/3-1/3).

Category name	Health	Business	Culture	Science	Sport
Health	58.0%	13%	4%	3.7%	21.3%
Business	4.6%	**73.5%**	5.3%	4.6%	12%
Culture	2.3%	10%	57.0%	0.7%	30%
Science	13.3%	5.3%	2.3%	59.1%	20%
Sport	2.0%	1.3%	3.6%	1.3%	**91.8%**

Table 5. Confusion Matrix results for cross validation, with no feature extraction (Leave-one-out)

The diagonals in tables 2-5 indicate higher classification performance for categories: Sport and Business than for the categories: Culture, Science, and health. Moreover, the leave-one-out experiment yields the best result by category as illustrated in Table 5 compared to the error rates reported in tables 2-4. Tables 2-5 revealed that error rates by category decrease from experiment to experiment. In other words, the error rates recorded in 1/3-2/3 experiment are higher than those in 1/2-1/2 experiment, those in 1/2-1/2 experiment are higher than those in 2/3-1/3 experiment, and those obtained in the 2/3-1/3 experiment are higher than those in the leave-one-out experiment. Thus, larger training sets yield higher accuracy when all the data set terms are used.

When investigating some of the misclassifications/confusions made by NB, we have noticed that misclassified documents, in fact, contain large number of words that are representative of other categories. In other words, documents that are known to belong to a category contain numerous words that have higher frequency in other categories. Therefore, these words have higher influence on the prediction that will be made by the classifier. For instance, the confusion matrix in Table 5 shows that 30% of Culture documents have been misclassified in the Sports category. The misclassified documents contain words that are more frequent in the Sports category such as جائزة (Arabic for prize and for trophy), بطل (Arabic for champion and for lead character), and تسجيل (Arabic for scoring and for recording).

5.2.2. Cross-validation, using feature selection

Feature selection techniques have been widely used in information retrieval as a means for coping with the large number of words in a document; a selection is made to keep only the more relevant words. Various feature selection techniques have been used in automatic text categorization; they include document frequency (DF), information gain (IG) (Tzeras and Hartman, 1993), minimum description length principal (Lang, 1995), and the χ^2 statistic. (Yang and Pedersen, 1997) has found strong correlations between DF, IG and the χ^2 statistic for a term. On the other hand, (Rogati and Yang, 2002) reports the χ^2 to produce best performance. In this paper, we use TF-IDF (a kind of augmented DF) as a feature selection criterion, in order to ensure results are comparable with those in (Yahyaoui, 2001).

TF-IDF (term frequency-inverse document frequency) is one of the widely used feature selection techniques in information retrieval (Yates and Neto, 1999). Specifically, it is used as a metric for measuring the importance of a word in a document within a collection, so as to improve the recall and the precision of the retrieved documents. While the TF measurement concerns the importance of a term in a given document, IDF seeks to measure the relative importance of a term in a collection of documents. The importance of each term is assumed to be inversely proportional to the number of documents that contain that term. TF is given by $TF_{D,t}$, and it denotes frequency of term t in document D. IDF is given by $IDF_t = \log(N/df_t)$, where N is the number of documents in the collection, and df_t is the number of documents containing the term t. (Salton and Yang, 1973) proposed the combination of TF and IDF as weighting schemes, and it has been shown that their product gave better performance. Thus, the weight of each term/root in a document is given by $w_{D,t} = TF_{D,t} * IDF_t$.

We have conducted five cross validation experiments based on TF-IDF. Experiments are based on selecting, in turn, 50, 100, 500, 1000, and 2000 terms that best represent the predefined 5 categories. We have repeated the experiments in Figure 3 for each number of terms. A summary of the results is presented in Table 6. The performance levels obtained are comparable to those obtained without feature selection. Figure 4 plots average categorization error rates versus the number of terms used for different trials.

Experiments #terms/roots	1/3-2/3	1/2-1/2	2/3-1/3	Leave-one-out
50	75.2(69.92,77.42)	64.88(60.32,68.4)	53.48(49.62,56.14)	36.9(0,100)
100	73.44(67.2,77)	62.58(59,66.7)	49.44(46.62,53.96)	33.7(0,100)
500	71.82(65.94,75.5)	60.32(55.9,64.24)	48.96(45.66,52.3)	33.16(0,100)
1000	69.54(64.06,72.12)	57.08(52.58,62.1)	46.96(42.84,50.76)	32.18(0,100)
2000	66.18(61.3,69)	53.96(46.9,66)	44.38(40.8,47.58)	31.22(0,100)
5000	67(62,69.9)	55(48.1,56.5)	46(42,49)	32.1(0,100)

Table 6. The overall error rate of NB in cross validation experiments using feature selection, in format: Avg(Min, Max)

Category	NB accuracy
Health	50%
Business	70%
Culture	40%
Science	60%
Sport	90%

Table 7. Classification accuracy on the evaluation set using Leave-one-out and TF-IDF with 2,000 roots/terms

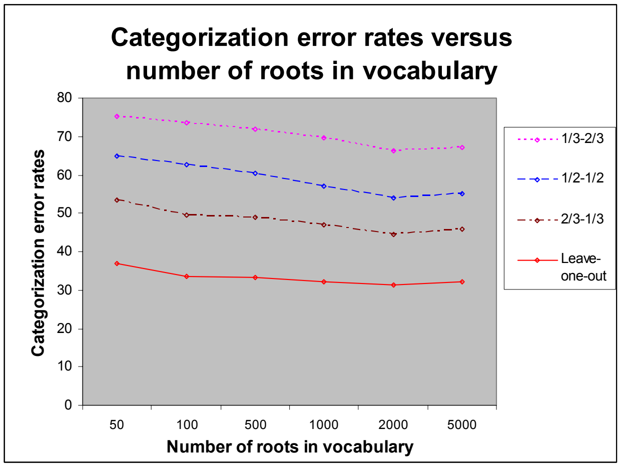

Figure 4. Categorization error rates versus number of terms.

5.3 Experiments using an evaluation set

Cross validation has been used to determine the average performance of NB for Arabic text categorization, and to design training sets that produce the best performance. This experiment, based on a separately and independently constructed evaluation set, is designed to evaluate the performance of NB on a set of documents that have never been submitted to the classifier. For this purpose, we further carefully collected manually 10 documents from Aljazeera.net for each of the 5 predefined categories. For each category, we have selected documents that best represent the variability in the category. We refer to this collection of documents as the evaluation set. This set is presented to the classifier for categorization.

For testing on the evaluation set, trained NB classifiers are used. For each category, we use the NB classifier that has been trained using the training set that produced the best category classification accuracy in cross validation experiments. In our case, we have used the whole set as a training set (1,500) represented by 2,000 terms since the best cross validation accuracy was obtained in leave-one-out experiment with 2,000 terms. Table 7 summarizes NB's performance results when tested using the evaluation set. The results obtained have shown higher performance for the Sports and the Business categories with a classification accuracy that is higher than 70%. The performance of other categories ranges from 40% to 60%. The average accuracy over all categories is 62%.

The results obtained in the evaluation set experiment are very consistent with the performance obtained in cross validation experiments.

6 Conclusions

To sum up, this work has been carried out to automatically classify Arabic documents using the NB algorithm, with the use of a different data set, a different number of categories, and a different root extraction algorithm from those used in (Yahyaoui, 2001). In this work, the average accuracy over all categories is: 68.78% in cross validation and 62% in evaluation set experiments. The corresponding performances in (Yahyaoui, 2001) are 75.6% and 50%, respectively. Thus, the overall performance (including cross validation and evaluation set experiments) in this work is comparable to that in (Yahyaoui, 2001). This offers some indication that the performance of NB algorithm in classifying Arabic documents is not sensitive to the Arabic root extraction algorithm. Future work will be directed at experimenting with other root extraction algorithms. Further improvement of NB's performance may be effected by using unlabeled documents; e.g., EM has been used successfully for this purpose in (Nigam et al., 200), where EM has increased the classification accuracy by 30% for classifying English documents. Two (English) document categorization algorithms have been reported to produce best results: Support Vector Machines (SVM) and AdaBoost. If the similarity between NB's performance for English and Arabic is any indication, SVM and AdaBoost should be the next candidates for application to Arabic Document categorization.

References

R. Al-Shalabi, and M. Evens, "A computational morphology system for Arabic," *In Workshop on Computational Approaches to Semitic Languages*, COLING-ACL98, 1998.

B. Cestink, "Estimating probabilities: A crucial task in machine learning," *Proceedings of the Ninth European Conference on Artificial Intelligence*, pp. 147--149, London, 1990.

K. Crammer and Y. Singer, “A Family of Additive Online Algorithms for Category Ranking,” *JMLR,* v. 3, pp. 1025-1058, Feb. 2003.

R. H. Creecy, B. M. Masand, S. J. Smith, and D. L. Waltz, “Trading mips and memory for knowledge engineering,” *Communication of the ACM*, Vol. 35, No. 8, pp. 48--64, August 1992.

M. El Kourdi, T. Rachidi, and A. Bensaid, "*A concatenative approach to Arabic word root extraction*," in progress, 2004.

Y.C. Fang, S. Parthasarathy and F. Schwartz, "Using clustering to boost text classification," *ICDM Workshop on Text Mining (TextDM'01)*, 2001.

Y. Houmame, *Towards an Arabic Information Retrieval System*, MS thesis, AlAkhawayn University, Morocco, 1999.

T. Joachims, *Learning to classify text using SVM*, Kluwer Academic Publishers, 2002.

K. Lang, "Newsweeder: Learning to filter netnews," *Proceedings of the Twelfth International Conference on Machine Learning,* 1995.

D. Lewis, M. Ringnette, "Comparison of two learning algorithms for text categorization," *Proceedings of the Third Annual Symposium on Document Analysis and Information Retrieval (SDAIR'94)*, 1994.

D. Lewis, Yiming Yang, Tony G. Rose, Fan Li, “A New Benchmark Collection for Text Categorization Research,” JMLR, v. 5, pp. 361-397, Apr. 2004.

T. Mitchell. Machine learning. McGraw Hill, 1997.

K. Nigam, A. K. McCallum, S. Thrun, and T.Mitchell, "Text classification from labeled and unlabeled documents using EM," *Machine Learning*, vol. 39, pp. 103--134, 2000.

T. Rachidi, O. Iraqi, M. Bouzoubaa, A. Ben Al Khattab, M. El Kourdi, A. Zahi, and A. Bensaid, “Barq: distributed multilingual Internet search engine with focus on Arabic language,” *Proceedings of IEEE Conf. on Sys., Man and Cyber., Washington DC, October 5-8, pp. , 2003.*

M. Rogati and Y. Yang. “High-performing feature selection for text classification,” ACM CIKM 2002.

Sakhr software company's website: www.sakhrsoft.com, 2004.

G. Salton and C. S. Yang, "On the specification of term values in automatic indexing", *Journal of Documentation*, Vol. 29, No. 4, pp. 351--372, 1973.

F. Sebastiani, “Machine learning in automated text categorization,” *ACM Computing Surveys*, v.34 n.1, p.1-47, March 2002.

K. Tzeras and S. Hartman, "Automatic indexing based on Bayesian inference networks," *Proc 16th Ann Int ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'93)*, pp. 22--34, 1993.

(Wiene and Pedersen, 1995) E. Wiener, J. O. Pedersen, and A. S. Zeigend, "A neural network approach to topic spotting," *Proceedings of the Fourth Symposium on Document Analysis and Information Retrieval (SDAIR'95)*, 1995.

M. Yahyaoui, "*Toward an Arabic web page classifier*," Master project. AUI. 2001.

Y. Yang, “An evaluation of statistical approaches to text categorization,” *Journal of Information Retrieval*, Vol. 1, Number 1-2, pp. 69--90, 1999.

Y. Yang and X. Liu, “A re-examination of text categorization methods,” *Proceedings of ACM SIGIR Conference on Research and Development in Information Retrieval* (SIGIR'99), pp 42--49, 1999.

R. B. Yates, and B. R. Neto, *Modern information retrieval*. Addison-Wesley ISBN 0-201-39829-X, 1999.

Yang, Y., Pedersen J.P. A Comparative Study on Feature Selection in Text Categorization *Proceedings of the 14th International Conference on Machine Learning*, pp. 412-420, 1997.