Journal of Computer Science 7 (1): 39-45, 2011
ISSN 1549-3636


# Statistical Bayesian Learning for Automatic Arabic Text Categorization

Bassam Al-Salemi and Mohd. Juzaiddin Ab Aziz
Department of Computer Science, Faculty of Information Technology,
University Kebangsaan Malaysia, Bangi, 43600, Selangor, Malaysia

**Abstract: Problem statement:** The rapid increasing of online Arabic documents necessitated applying Text Categorization techniques that are commonly used for English language to categorize them automatically. The complex morphology of Arabic language and its large vocabulary size make applying these techniques directly difficult and costly in time and effort. **Approach:** We have investigated Bayesian learning models in order to enhance Arabic ATC. Three classifiers based on Bayesian theorem had been implemented which are Simple Naïve Bayes (NB), Multi-variant Bernoulli Naïve Bayes (MBNB) and Multinomial Naïve Bayes (MNB) models. TREC-2002 Light Stemmer was applied for Arabic stemming. For text representation, Bag-Of-Word and character-level n-gram with the length 3, 4 and 5 are used. In order to reduce the dimensionality of feature space, the following feature selection methods: Mutual Information, Chi-Square statistic, Odds Ratio and GSS-coefficient were used. **Conclusion:** MBNB classifier outperformed both of NB and MNB classifiers. BOW representation leads to the best classification performance; nevertheless, using character-level n-gram leads to satisfying results for Arabic ATC based on Bayesian learning. Moreover, the use of feature selection methods dramatically increases the categorization performance.

**Keywords:** Arabic Text Categorization, Bayesian Learning, Feature Selection, Automatic Text Categorization, Multinomial Naïve Bayes, Multivariate Bernoulli Naïve Bayes, Odds Ratio (OR), Information Gain (IG), Feature Selection (FS), Mutual Information (MI).

## INTRODUCTION

Automatic Text Categorization (ATC) is the task of assigning a given document to its predefined category automatically. In recent years, using the computer in our life leads to increase the number of electronic documents and digital information. As a result, ATC has become one of the most powerful techniques for organizing the data. Instead of using the classical models of text classification that consist of a set of logical rules defined manually, Machine Learning (ML) approach had been applied widely to classify the texts automatically with high accuracy (Sebastiani, 2002). The most common Supervised ML algorithms are Statistical Learning algorithms, which provide a probability that a given document being assigned to particular classes based on probabilistic model (Kotsiantis, 2007; Sebastiani, 2002; Yang and Pedersen, 1997). Bayesian learning model is statistical learning model that based on Bayesian theorem of the independency of feature terms given the classification. Naïve Bayes (NB), Multivariate Bernoulli Naïve Bayes (MBNB), Multinomial Naïve Bayes (MNB) (Eyheramendy *et al*., 2003; Kotsiantis, 2007; McCallum and Nigam, 1998; Schneider, 2003; Mendez *et al*., 2008; Yang and Pedersen, 1997) are probabilistic models, which all apply Bayesian theorem while the way of computing the probability is different.ML approach divided into two phases; training phase and test phase. For the training phase, a set of documents of the collected corpus (called training set) is used to build the classifier by allotment a subset of the training set for each category and process them by several Information Retrieval (IR) techniques to extract a set of features used as characteristics for each category. In the test phase, the remainder of the corpus (called test set) will be used to test and evaluate the performance of the classifier by classifying the documents under each category as unseen documents and then compare the estimated categories to the pre-defined ones to measure the classification performance. Typically, there are two representation methods to represent the text as a set of features; Bag-Of-Word (BOW) by using single words or phrases as features and n-gram by using sequence of words (Word-Level n-gram) or characters (Character Level n-gram) of the length n (El-Kourdi *et al*., 2004). One problem arises of building ATC system is handling the huge number of

**Corresponding Author:** Bassam Al-Salemi, Department of Computer Science, Faculty of Information Technology, University Kebangsaan Malaysia, Bangi, 43600, Selangor, Malaysia

features, which can easily reach orders of tens of thousands (Al-Harbi *et al*., 2008; Eyheramendy *et al*., 2003). For reducing the feature space dimension, many IR techniques have been applied, such as Stemming, Stop-words Removal and Feature Selection (FS). FS techniques such as Mutual Information (MI), Chi-Square Statistic (CHI), Information Gain (IG), GSS Coefficient (GSS) and Odds Ratio (OR) used to reduce the dimensionality of feature space by eliminating the features that are considered irrelevant for a particular category (Al-Harbi *et al*., 2008; Duwiri, 2007; Forman, 2003; Fragoudis *et al*., 2005; Galavotti *et al*., 2000).

The main aim of this study is to evaluate and enhance the performance of the following Bayesian-based classifiers: NB, MBNB and MNB for Arabic ATC and analysis the effect of using the following FS methods: CHI, OR, MI and GSS on the classification performance.

**Related work:** Many Bayesian learning and other statistical learning models have been applied for ATC. The bulk of ATC work has been devoted for English and other Latin language. Concerning Bayesian learning, McCallum and Nigam (McCallum and Nigam, 1998) have carried out an analysis study of MNB and MBNB performance for English ACT. Their results proved that MNB outperforms MNB. In addition, MNB can perform well when the feature space size decreased. Another study conducted by Schneider (Schneider, 2003) of using MBNB and MNB for spam filtering. The findings confirmed that MNB outperforms MBNB.

Unlike English language, a limited number of studies had been done for Arabic ATC(Al-Harbi *et al*., 2008; Darwish and Oard, 2002; Duwiri, 2006; 2007; Harrag *et al*., 2009; Kanaan *et al*., 2009; Khreisat, 2009; Mesleh, 2008; 2007). Among all of them only (Duwiri, 2006; 2007; Kanaan *et al*., 2009) used Bayesian learning model. However, they employed the simple NB, while MNB and MBNB, which we will investigate in this work, may achieve better.

Arabic language consists of 28 letters and unlike English, it written from right to left.In addition, Arabic has a complex morphology (El-Kourdi *et al*., 2004; Haraty and Ariss 2007). For that reasons, applying ACT techniques for Arabic is more complicated than that for English.

## METHODS AND MATERIALS

**Preliminary**: Let $X = \{x_i: i = 1,\ldots,n\}$ be a finite set of documents and let $Y = \{y_i: j = 1,\ldots,m\}$ be a finite set of labels such that each document $x_i \epsilon X$ belongs to a class label $y_i \epsilon Y$, given a set of training examples, $S = \{(x_i,y_i), i = (1,\ldots,n)\}$. The Bayesian learning task is to build from the training set a probabilistic model capable of estimating the conditional probability of the class y given an example x,p(y|x), for all possible values of y and x.

**Arabic Text Pre-processing:** Like in any ACT system, the first step is pre-processing the plain texts. For Arabic texts, text pre-processing usually involves the following: removing punctuation marks, diacritics and non-Arabic letters, excluding the words with length less than three and eliminating stop-words (Khreisat, 2009; Larkey *et al*., 2007). In this study, Arabic TREC-2002 Light Stemmer (Darwish and Oard, 2002) is employed to return the words to their stems by removing the most frequent suffixes and prefixes.

**Feature Selection:** Given a category $y_i \epsilon Y$ and a feature term t belongs to one or more documents in X. Let A denotes to the number of times t presents in $y_i$, B is the number of times t presents without $y_i$, C is the number of times t absents in $y_i$, D is the number of times t absents without $y_i$ and n is the training set size. CHI, MI, OR and GSS methods compute the score of t belongs to $y_i$ as the following:

$$CHI(t, y_i) \approx \frac{n(AD - CB)}{(A + C)(B + C)(A + B)(C + D)} \quad (1)$$

$$MI(t, y_i) \approx \log_2 \log_2 \frac{n \times A}{(A + C)(A + B)} \quad (2)$$

$$OR(t, y_i) \approx \frac{AD}{CB} \quad (3)$$

$$GSS(t, y_i) \approx \frac{AD - CB}{n^2} \quad (4)$$

Max score of each FS function calculated as

$$Max_{score} = \max_{i=1,..,m} FS(t, y_i) \quad (5)$$

$Max_{score}$ returns the appropriate category that t belong to.

**Classifiers:** Given a document x represented as a set of feature terms $x = \{t_i: i = 1,..,|x|\}$ and a category y. The conditional probability of y given x,p(y|x), (called *posterior probability*) estimated as follows:

$$p(y|x) = p(y|t_1,\ldots,t_k) = p(y)\prod_{i=1}^{|x|} p(t_i \mid y) \quad (6)$$

Thus, the *Bayes optimal classifier*, the classifier that achieve the minimum error, is chosen according to:

$$y^* = \arg\max_y \left\{ p(y)\prod_{i=1}^{|x|} p(t_i \mid y) \right\} \quad (7)$$

Therefore, the document x is classified to the category $y^*$.

Particularly, if we denote to the number of documents under y which contain $t_i$ as $n_{yi}$ and the total number of documents under y as $n_y$. Then, the probability $p(t_i|y)$ is estimated using *Laplac prior* (Chen *et al*., 2009) as:

$$p(t_i \mid y) = \frac{1 + n_{yi}}{m + n_y} \quad (8)$$

where, m is the number of all categories and p(y) is the probability of y computed as:

$$p(y) = \frac{\text{Number of documents in y}}{\text{Total number of documents}} \quad (9)$$

Bayesian classifier introduced so far is the simple form of Naïve Bayes, for simplicity we call it NB.

**MBNB:** Suppose that, the feature set that extracted from the training set is $T = \{t_1,\ldots,t_k\}$. In MBNB, each document x is represented as a binary vector $\vec{v} = < v_1,\ldots,v_k >$ in which $v_i = 1$ if $t_i$ occurs in the document x (at least once), or $v_i = 0$ otherwise. Thus, each document x is seen as a result of k Bernoulli trials, where for each trial we decide whether or not $t_i$ occurs in x. Under the naïve Bayes assumption that the probability of each word occurring in a document is independent of other words given the class label, the probability p(x|y) is computed as a simple product:

$$p(x|y) = p(\vec{v}|y) = \prod_{i=1}^{k} p(t_i|y)^{v_i} \cdot \left(1 - p(t_i|y)\right)^{1-v_i} \quad (10)$$

Therefore, the maximum posteriori classifier is constructed as:

$$y^* = \arg\max_y \left\{p(y).p(\vec{v}|y)\right\} \quad (11)$$

where, $p(t_i|y)$ and p(y) come from Equation (8) and (9) respectively.

**MNB:** MNB represents the document $x = \{t_1,\ldots,t_{|x|}\}$, as a vector $\vec{v} = < v_1,\ldots,v_{|x|} >$, where $v_i$ is the number of occurrence of $t_i$ in x. The probability p(x|y) computed as the multinomial distribution:

$$p(x|y) = p(\vec{v}|y) = p(|x|).|x|!\prod_{i=1}^{|x|} \frac{p(t_i \mid y)^{v_i}}{v_i!} \quad (12)$$

While |x| does not depend on the category y, then, there is no need to calculate p(|x|) and |x|! Schneider (2004). Moreover, if we denote to the number of occurrences of $t_i$ in category y as $n_{yi}$ and the number of the terms in category y as $n_y$. So, the probability $p(t_i|y)$ is estimated by means of *Laplac prior* as:

$$p(t_i \mid y) = \frac{1 + n_{yi}}{n + n_y} \quad (13)$$

where, n is the total number of all documents and p(y) computed as:

$$p(y) = \frac{\text{Number of selected terms in y}}{\text{feature set size}} \quad (14)$$

**Performance measures:** The effectiveness of ACT system can be measured by sorting the categorization result into the following:

For each category y, suppose that the classifier predictions are summed up as follows: True Positive (TP) refers to the set of documents that assigned correctly to y, False Positive (FP) refers to the set of documents incorrectly assigned to y, False Negative (FN) refers to the set of documents incorrectly not assigned to y and True Negative (TN) refers to the set of documents correctly not assigned to y.

**Precision and recall:** Precision (*p*) and Recall (*r*) of a category y defined as:

$$p = \frac{TP}{TP + FN} \quad (15)$$

$$r = \frac{TP}{TP + FP} \quad (16)$$

**F1-measure:** F1-measure is the most widely measure used to measure the classification performance and computed as the harmonic mean of p and r taken the form:

$$F1 = \frac{2p.r}{p + r} \quad (17)$$

**Macroaveraged-F1 (Macro-F1):** Macro-F1 computed as the arithmetic average over F1-measure of all categories:

$$\text{Macro - F1} = \frac{1}{m}\sum_y F1 \quad (18)$$

## EXPERMENTS AND RESULTS

The dataset used in this study is in-house collections of Arabic news consists of 3,172 documents and fill into the following categories: Arts, Economy, Politics and Sport. Dataset divided into 1,732 documents for training and 1,440 documents for test. Table 1 shows how the dataset divided for training and test per category.

The first step is pre-processing the plain texts. The pre-processing involves tokenization, normalization, stop-words removal and stemming. For text representation, we used character-level n-gram of length 3, 4 and 5 and stemmed-words. After representing the text, we extract four different features sets, one for each representation methods. Then, we employed FS methods for reducing the features dimension. The FS methods employed in our study are the following: CHI, MI, OR and GSS. Table 2 shows the impact of using FS methods for reducing the number of selected features as stemmed-words. Then, we built and trained the following Bayesian learning models: NB, MBNB and MNB.

For three feature representation methods, four feature selection techniques and three classifiers, the number of experiments carried out is 36 different experiments in which the number of experiments for each classifier is 12 experiments.

In each experiment, we evaluated the performance of each classifier on the test set using different number of the top most frequent terms in each feature set. The given numbers of the top selected features are 200, 400, 600, 800, 1,000 and 1,200 features.

**Results using 3-gram representation**: Fig. 1 shows the Macro-F1 results using 3-gram representation. It is clear that MNB classifier achieved the best performance using GSS method. The best Macro-F1 result obtained by MNB is (0.912) when the number of feature terms is 1,000 or 1,200. The second Macro-F1 result is achieved also by MNB with OR, which is (0.907) occurred when the number of selected features is 1,200 feature. MBNB comes after MNB in which the best result obtained is (0.902) using OR when the number of top terms is 1,000 or 1,200. However, unlike NB and MNB, MBNB performs well with small number of features. The best Macro-F1 results obtained by NB is (0.897) using GSS when the number of top terms is 1,000 or 1,200. Concerning FS methods, CHI leads to the worst performance overall, while GSS approximately leads to the best performance.

Table 1: The categories and their training and test set

| | Art | Politics | Economic | Sport |
|---|---|---|---|---|
| Training set | 414 | 430 | 543 | 345 |
| Test set | 360 | 360 | 360 | 360 |

Table 2: Stemmed-words feature set size for each category before and after applying FS methods.

| FS method | Art | Politics | Economic | Sport |
|---|---|---|---|---|
| Without | 5360 | 5039 | 4065 | 4182 |
| CHI | 2215 | 1614 | 1827 | 1533 |
| GSS | 2280 | 1719 | 1785 | 1399 |
| MI | 2289 | 1513 | 1841 | 1546 |
| OR | 2285 | 1545 | 1831 | 1528 |

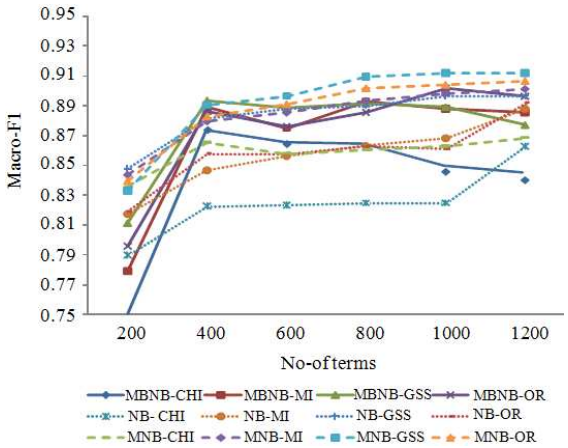


Fig. 1: Macro-F1 resultsusing 3-gram representation

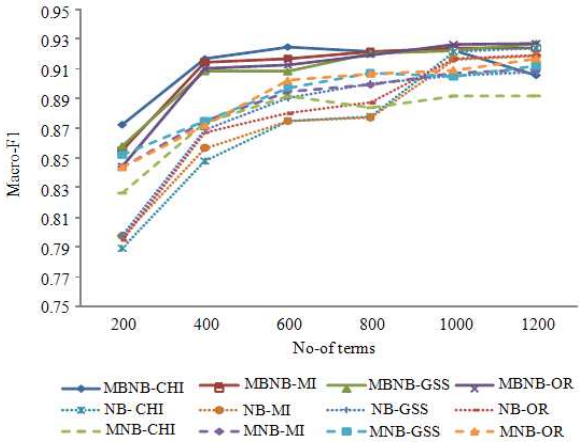


Fig. 2: Macro-F1 resultsusing 4-gram representation

**Results using 4-gram representation**: Fig. 2 shows that MBNB achieved the best performance overall using 4-gram representation. The best Macro-F1 achieved by MBNB is (0.927) when the number of features is 1,200 selected by OR or GSS. MNB and NB obtain the lowest performance when the number of features less than 1,000, while they achieve better when

the number of features over than 1,000. The best result obtained by NB is (0.924) with 1,200 features selected by CHI. MBNB achieved better than NB in average, while the best Macro-F1 achieved is (0.924) when the number of features is 1,200 selected by OR method.

**Results using 5-gram representation:** From Fig. 3, it is clear that MBNB achieved the best performance over all. However, increasing the 5-gram features more than 400 is not effective in large. NB and MNB are performing well when the number of features is more than 1,000. The best Macro-F1 achieved by MBNB is (0.934) occurred by using 1,000 features selected by OR method.MNB classifier achieved better than NB when the number of features less than 1,000 and after that NB outperforms MNB; however, increasing the number of features enhances the performance of both MNB and NB. The best Macro-F1 results achieved by NB and MNB are (0.929) and (0.891) obtained when the number of features are 1,000 and 1,200 respectively.

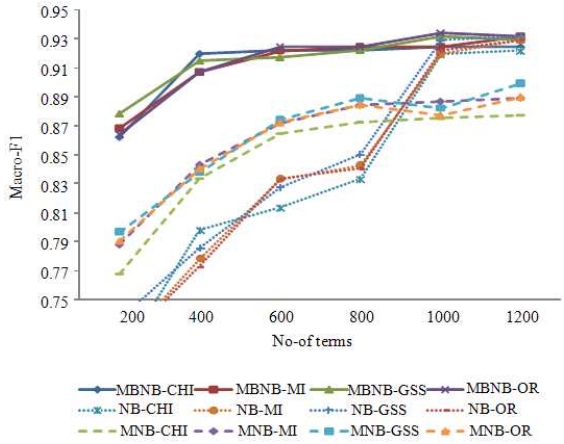


Fig. 3: Macro-F1 resultsusing 5-gram representation

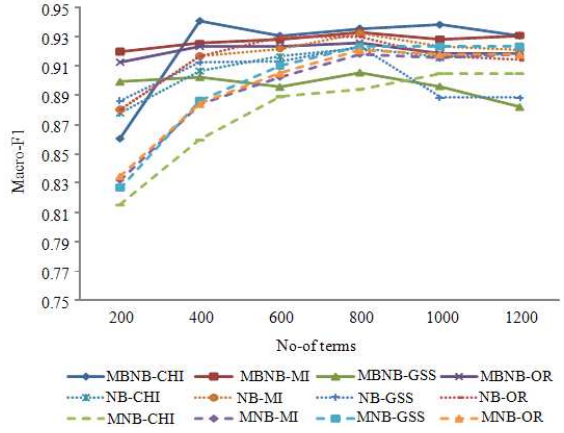


Fig. 4: Macro-F1 resultsusing stemmed-words

**Results using BOW representation (stemmed-words):** Fig. 4 shows that MBNB classifier achieved well using small number of top features, while MNB is performing better when the number of features increased. However, MBNB outperforms both MNB and NB in general. The best Macro-F1 result achieved by MBNB is (0.941) using 400 features, selected by CHI method. However using MI to select the features for MBNB leads to (0.933) Macro-F1, when the number of top features is 800. NB outperforms MNB when the number of features less than 800 features with the best Macro-F1 value (0.933) using 800 features, selected by MI.

## DISCUSSION

MBNB outperformed MNB and NB overall. The reason behind that is the number of extracted features from the dataset is not too large. Moreover, the dataset is small and balanced. These findings are dealing with McCallum and Nigam (1998) findings. In their study that conducted on Bayesian learning for ATC, they pointed out that MBNB almost achieved accurate performance with small number of features less than 1,000 features and when the dataset is balanced. However, in our study MBNB outperforms MNB for the same reason mentioned above.

Furthermore, MNB can outperform both of NB and MBNB when the number of features is extremely large and when the features occurred many times in the training data. For instance, using 3-gram representation leads to increase the features occurrences in the training data and as a result, MNB achieved the best performance.

In addition, using the stemmed-words as features leads to the best performance among all the used text representation techniques, nevertheless using character level n-gram for Bayesian learning models leads to accepted results; however, 3-gram representation leads to the poorest performance.

Table 3: The best choosing of FS methods that leads to the best performance

| Classifier | Feature type | Number of top features | | |
|---|---|---|---|---|
| | 200 | 400 | Over | |
| MBNB | 3-gram | GSS | GSS | MI |
| | 4-gram | CHI | CHI | OR |
| | 5-gram | GSS | CHI | OR |
| | BOW | MI | GSS | OR |
| NB | 3-gram | GSS | GSS | MI |
| | 4-gram | GSS | GSS | CHI |
| | 5-gram | GSS | CHI | GSS |
| | BOW | GSS | MI | OR |
| MNB | 3-gram | MI | GSS | MI |
| | 4-gram | CHI | GSS | OR |
| | 5-gram | GSS | MI | GSS |
| | BOW | OR | GSS | GSS |

## CONCLUSION

In this study, we investigated the use of Bayesian learning for Arabic Text Categorization. Two representation type were used for representing the text and four feature selection methods were investigated to reduce the feature space dimensionality. The experimental results on a collection of Arabic news proved that MBNB outperforms both MNB and NB overall and using BOW representation leads to the best performance. Furthermore, our findings verified that using n-gram is not limited on the distance-based classification models. In our experiment, we have investigated character level n-gram to represent Arabic texts for ATC based on Bayesian learning and it leads to accepted results. In addition, we have analyzed the relevance of choosing an appropriate feature selection method with the size and type of the features and their effectiveness on each classifier performance, (Table 3) sums up these findings.The bestMacro-F1obtained over all is (0.941), achieved by MBNB when the number of features is 400, represented by BOW and selected by CHI feature selection method.

In the future, we will expand the number of Arabic categories to cover the most common categories and we will include the other Bayesian learning classifiers that were not mentioned in this study.

## REFERENCES


Al-Harbi, S., A. Almuhareb, A. Al-Thubaity, M.S. Khorsheed and A. Al-Rajeh, 2008. Automatic Arabic Text Classification. 9es journées internationales d'analyse statistique des données textuelles, JADT, 08: 77-83.

Chen, J., H. Huang, S. Tian and Y. Qu, 2009. Feature Selection For Text Classification With Naïve Bayes. Expert systems with applications. Int. J., 36: 5432-5435. DOI: 10.1016/j.eswa.2008.06.054

Darwish, K. and D.W. Oard, 2002. CLIR Experiments at Maryland for TREC-2002: evidence combination for Arabic-English retrieval. Proceedings of the 11th Text Retrieval Conference. (TRC'02), Citrseerex Publisher, Pennsylvania State, pp: 703-710.

Duwiri, R.M., 2006. Machine learning for Arabic text categorization. J. Am. Soc. Inform. Sci. Technol., 57: 1005-1010. DOI: 10.1002/asi.20360

Duwiri, R., 2007. Arabic Text categorization. Int. Arab J. Inform. Technol., 4: 125-131.

El-Kourdi, M., A. Bensaid and T. Rachidi, 2004. Automatic Arabic document categorization based on the Naïve Bayes Algorithm. Proceedings of the Workshop on Computational Approaches to Arabic Script-Based Languages, (CAASL' 04), Stroudsburg, PA, USA, pp:51-58.

Eyheramendy, S., D. Lewis and D. Madigan, 2003. On the naive bayes model for text categorization. Citrseerex.

Forman, G., 2003. An extensive empirical study of feature selection metrics for text classification. J. Mach. Learn. Res. ACM, 3: 1289-1305. DOI: 10.1162/153244303322753670

Fragoudis, D., D. Meretakis and S. Likothanassis, 2005. Best Terms: An efficient feature-selection algorithm for text categorization. Knowl. Inform. Syst., 8: 16-33. DOI: 10.1007/s10115-004-0177-2

Galavotti, L., F. Sebastiani and M. Simi, 2000. Experiments on the use of feature selection and negative evidence in automated text categorization. Res. Adv. Technol. Digital Libraries, 1923: 59-68. DOI: 10.1007/3-540-45268-0_6

Haraty, R.A. and O.E.Ariss, 2007. CASRA+: A Colloquial Arabic Speech Recognition Application. Am. J. Applied Sci., 4: 23-32. DOI: 10.3844/.2007.23.32

Harrag, F., E. El-Qawasmeh and P. Pichappan 2009. Improving Arabic text categorization using decision trees. Proceedings of the 1st International Conference Networked Digital Technologies, July, 28-31, IEEE Xplore Press, Ostrava, pp: 110-115. DOI: 10.1109/NDT.2009.5272214

Kanaan, G., R. Al-Shalabi, S. Ghwanmeh and H. Al-Maadeed, 2009. A comparison of text-classification techniques applied to Arabic Text. J. Am. Soc. Inform. Sci. Technol., 60: 1836-1844. DOI:10.1002/ASI.20832

Khreisat, L., 2009. A machine learning approach for Arabic text classification using N-gram frequency statistics. J. Inform., 3: 72-77.

Kotsiantis, S.B., 2007. Supervised machine learning: A review of classification techniques. Informatica, 31: 249-268.

Larkey, L., L. Ballesteros and M. Connell, 2007. Light stemming for Arabic information retrieval. Arabic Comput. Morphol., 38: 221-243. DOI: 10.1007/978-1-4020-6046-5_12

McCallum, A. and K. Nigam, 1998. A Comparison Of Event Models For Naive Bayes Text Classification. Proceeding of the AAAI-98 Workshop on Learning for Text Categorization. (WLTC'98), Publisher Citeseer, Pennsylvania State, pp: 41-48.

Mendez, J.R., I. Cid, D. Glez-Peña, M. Rocha and F. Fdez-Riverola, 2008. A comparative impact study of attribute selection techniques on naïve bayes spam filters. Lect. Notes Comp. Sci., 5077: 213-227, DOI: 10.1007/978-3-540-70720-2_17

Mesleh, A.W. 2008. Support vector machines based Arabic language text classification System: Feature Selection Comparative Study. Proceedings of the 12th WSEAS International Conference on Applied Mathematics, Dec. 29-31, World Scientific and Engineering Academy and Society, Wisconsin, USA., pp: 228-233.

Mesleh, A.M.A., 2007. Chi square feature extraction based SVMs Arabic language text categorization system. J. Comput. Sci., 3: 430-435. DOI:10.3844/JCSSP.2007.430.435

Sebastiani, F., 2002. Machine learning in automated text categorization. ACM Comput. Surveys (CSUR), 34: 1-47. DOI:10.1145/505282.505283

Schneider, K., 2003. A comparison of event models for naive bayes anti-spam e-mail filtering. Proceeding of the 10th Conference on European Chapter of the Association for Computational Linguistics, (ECACL' 03) ACM, Budapest, Hungary, pp: 307-314. DOI: 10.3115/1067807.1067848

Yang, Y. and J. Pedersen, 1997. A Comparative Study on Feature Selection in Text Categorization. Proceedings of the Fourteenth International Conference on Machine Learning (ICML'97), Morgan Kaufmann Publishers Inc, CA, USA., pp: 412-420.