Neural Networks for the Automation of Arabic Text Categorization

Saleh M. AlSaleem
King Saud University
Riyadh, Saudi Arabia

ABSTRACT: *In this paper, we compare and investigate Naïve Bayesian Method (NB), K-Nearest Neighbor along with Neural Network method on different Arabic data sets. The bases of our comparison are the most popular text evaluation measures. The Experimental results against different Arabic text categorization data sets reveal that NB categorizer outperformed both k-NN and NN algorithms with regard to F1, Recall and Precision measures.*

Keywords: Text Categorization, Neural Network, Naïve Bayesian, Information Retrieval, K-Nearest Neighbor, Machine Learning

Received: 10 March 2013, Revised 9 April 2013, Accepted 18 April 2013

1. Introduction

Text categorization (*TC*) (also known as text classification or topic spotting) is defined as the possibility of automatically sorting a set of documents into categories from a predefined set. This task has numerous and extensive applications, including automated indexing of research articles according to predefined thesauri of technical terms, filing patents into patent directories, selective dissemination of information to information consumers, automated population of hierarchical catalogues of Web resources, spam filtering, identification of document genre that could be implemented in news articles and documents, authorship attribution, survey coding, and even automated grading of some text-based questions in educational environments.

Automated text classification is exceedingly attracting more attention due to the fact that it frees organizations from the need of manually organizing document bases, which can be too expensive, or simply not feasible given the time constraints of the application or the number of documents involved.

The accuracy of modern text classification systems is proving daily that it rivals that of trained human professionals, thanks to a combination of information retrieval (*IR*) technology and machine learning (*ML*) technology.

(*TC*) refers to the process of assigning a category or some categories among predefined ones to each document, automatically. A system that performs text categorization aims to assign appropriate labels (or categories) from a predefined classification scheme to incoming documents. In this paper we focus on a single label assignment.

Typical machine learning based approaches to text categorization are *K* Nearest Neighbor, Naïve Bayes, Support Vector Machine, and Neural Network. They are used not only for text categorization, but also for any pattern classification problem, such as image classification, protein classification, and character recognition. Although there are other approaches than the named approaches, the four approaches are most typical and popular. In section II, we will present previous cases of applying the three approaches, namely *K* Nearest Neighbor, Naïve Bayes, and Support Vector Machine to text categorization.

The aim of this paper is to present and compare results obtained against Saudi Newspapers (*SNP*) Arabic text collections [7] using Naïve Bayesian algorithm, *K*-Nearest Neighbor algorithm and Neural Network algorithm. The bases of our comparison of the *NB*, *KNN*, and *NN* are the most popular text evaluation measures (*F*1, Recall, and Precision) [21]. In other worlds we want to determine the categorizer that produces the best results concentrating more on *NN* and trying to explain the meaning of the results achieved. To the best of the author's knowledge, no such comparison have been conducted against *SNP* data collections using *NB*, *KNN*, and *NN* and evaluated using Recall, *F*1 and Precision measures.

This paper is organized as follows: section II gives an overview of related work. Section III introduces the problems of *TC*. Section IV presents results and discussion. Finally conclusion and future work are given in section V.

2. Related Work

Since *TC* stands at the cross junction to modern information retrieval and machine learning, several research papers have focused on it but each of which has concentrated on one or more issues related to such task. In last decade, there are many previous works conducted on Arabic *TC*. For instance, [11] compared between Manhattan distance and Dice measures using N-gram frequency statistical technique against Arabic data sets collected from several online Arabic newspaper websites. The results showed that N-gram using Dice measure outperformed Manhattan distance. The author's of [17] presented results using statistical methods such as maximum entropy to cluster Arabic news articles; the results derived by these methods were promising without morphological analysis.

In [10], *NB* was applied to classify Arabic web data; the results showed that the average accuracy was 68.78%.

[4] Used Maximum Entropy for *TC* on Arabic data sets, the results revealed that the average F-measure increased from 68.13% to 80.41% using pre-processing techniques (normalization, stop words removal, and stemming).

The algorithm developed by [5] has outperformed other presented text classification algorithms, i.e. [10], [4], [17], and [15] Categorizer with regards to F-measure results.

[12] Used three classification algorithms, namely *SVM*, *KNN* and *NB*, to classify 1445 texts taken from online Arabic newspaper archives. The compiled texts were classified into nine classes: Computer, Economics, Education, Engineering, Law, Medicine, Politics, Religion and Sports. Chi Square statistics was used for feature selection. [12] Discussed that "*Compared to other classification methods, our system shows a high classification effectiveness for Arabic data set in terms of F-measure* (*F* = 88.11)".

In [20], the authors investigated different variations of Vector Space Model using KNN algorithm, these variations are Cosine coefficient, Dice coefficient and Jacaard coefficient, using different term weighting approaches. The average F1 results obtained against six Arabic data sets indicated that Dice based *TF*. IDF and Jaccard based TF.IDF outperformed Cosine based TF.IDF, Cosine based WIDF, Cosine based ITF, Cosine based log (1 + *tf*), Dice based WIDF, Dice based ITF, Dice based log (1 + *tf*), Jaccard based WIDF, Jaccard based ITF, and Jaccard based log (1 + *tf*).

In [3], *NB* and *KNN* were applied to classify Arabic text collected from online Arabic newspapers including Al-Jazeera, Al-Nahar, Al-hayat, Al-Ahram, and Al-Dostor. The results show that the NB classifier outperformed KNN base on Cosine coefficient with regards to macro F1, macro recall and macro precision measures.

Finally, in [19] investigate *NB* algorithm based on Chi Square features selection method. The experimental results compared against different Arabic text categorization data sets provided evidence that feature selection often increases classification accuracy by removing rare terms.

3. TECT Categorization Problem

TC, also known as text classification, is the task of automatically sorting a set of documents, according to a predefined set, into categories (or classes, or topics). Tasks of this sort are related to the communities of *IR* and *ML*. The focus on Automated Text Classification tools is due to the fact that organizations can make good use of the saved time, and saved effort which was expensive or sometimes even not feasible given the huge amounts of data involved with limitless constraints forced on its classification.

For *IR* researchers, this interest is one particular aspect of a general movement towards leveraging user data for taming the inherent subjectivity of the *IR* task, i.e. taming the fact that it is the user, and only the user, who can say whether a given item of information is relevant to a query issued to a Web search engine, or to a private folder in which documents should be filed according to content. Wherever there are predefined classes, documents manually classified by the user are often available; as a consequence, this latter data can be exploited for automatically learning the (extensional) meaning that the user attributes to the classes, thereby reaching levels of classification accuracy that would be unthinkable if this data were unavailable.

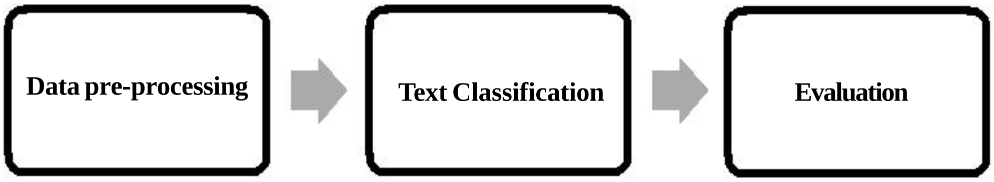
For *ML* researchers, this interest is due to the fact that *IR* applications prove an excellent and challenging benchmark for their own techniques and methodologies, since *IR* applications usually feature extremely high-dimensional feature spaces and provide data by the truckload. In the last five years, this has resulted in more and more *ML* researchers adopting *TC* as one of their benchmark applications of choice, which means that cutting-edge *ML* techniques are being imported into *TC* with minimal delay from their original invention.

For application developers, this interest is mainly due to the enormously increased need to handle larger and larger quantities of documents, a need emphasized by increased connectivity and availability of document bases of all types at all levels in the information chain. But this interest is also due to the fact that *TC* techniques have reached accuracy levels that rival the performance of trained professionals, and these accuracy levels can be achieved with high levels of efficiency on standard hardware/software resources. This means that more and more organizations are automating all their activities that can be cast as *TC* tasks.

TC encompasses several applications such as automated indexing of scientific articles according to predefined thesauri of technical terms, filing patents into patent directories, selective dissemination of information to information consumers, automated population of hierarchical catalogues of web resources, spam filtering, identification of document genre, authorship attribution, survey coding and even grading of educational text-based material.

TC problem can be defined according to [18] as follows: let G denote the collection of categories which contain $\{g_1, g_2, …..g_n)$, let D denote the collection of documents and Q is an incoming text. Moreover, let R denote the set of classifiers for $D \times Q \rightarrow G$ each document d & D is assigned a single class g that belongs to G. The goat is to find a classifier h & H that maximizes the probability that $r(d) = G$ for each test case (d, g).

Generally, *TC* task goes through these basic steps:



Data pre-processing phase is to prepare the text documents and make it ready for classifier training process. Following that, the text classifier is constructed and tuned by deploying a text learning approach against the training data set. The final step is the evaluation of the text classifier by some evaluation measures i.e. Recall, Precision, etc. The following two sub-sections are devoted to the discussion of the main phases of the *TC* problem in relation to the data utilized in this paper.

3.1 Data Pre-Processing on Arabic text documents

The data utilized in our experiments is The Saudi Newspapers (SNP) [7], the data set consist of 1400 Arabic documents of different lengths that would fall into any of seven categories. The categories are (Medical “طبية”, Economics “الإقتصادية”, Entertainment “ترفيهية”, Information Technology “المعلوماتتكنولوجيا”, Local News “اخبار محلية”, International News “عالميةاخبار”, Sport “الرياضة”, Table 1 represents the number of documents in each category.

Arabic language is highly inflectional and derivational language. It varies drastically from English language text. Accordingly, monophonical analysis of Arabic language text documents is a complex task. Moreover, Arabic scripts might have works that contain vowels represented by diacritics which are usually left out in the text. Above that, Arabic scripts do not use capitalization for proper nouns. This could create some ambiguity in the text [6]. It will require several processing steps. In the Arabic data set we use, each document file was saved in a separate file within the corresponding category's directory.

Category Name	Number of Documents
Medical	190
Economics	145
Entertainment	180
Information Technology	190
Local News	230
International News	245
Sport	220
Total	1400

Table 1. Number of Documents per Category

In the first pre-processing steps, Arabic data set is converted into a form that is suitable for classification algorithm In this phase, we have followed [1], [2], [10] data format and processed the Arabic documents according to the following steps:

1. Each article in the Arabic data set is processed to remove the digits and punctuation marks.

2. We have followed [16] in the normalization of some Arabic letters such as the normalization of (hamza (إ) or (أ)) in all its forms to (alef (ا)).

3. All the non-Arabic texts were filtered.

4. Arabic function words were removed. The Arabic function words (stop words) are the words that are not useful in IR systems e.g. The Arabic prefixes, pronouns, and prepositions.

3.2 Approaches to Text Categorization

The current sub-section covers three approaches to text categorization: NB, K-NN and NN. NB is a simple probabilistic classifier based on Baye's theorem. It is powerful, easy and language independent method. When the NB classifier is applied on the TC problem we use equation 1.

$$p\ (class \mid document) = \frac{p\ (class)\ p\ (document \mid class)}{p\ (document)} \qquad (1)$$

Where:
P(*class* | *document*): is the probability of class given a document, or the probability that a given document *D* belongs to a given class *C*. *P* (*document*):The probability of a document, we can notice that *p* (*document*) is a Constance divider to every calculation, accordingly, we can ignore it. *P* (*class*): The probability of a class (or category), we can compute it from the number of documents in the category divided by documents number in all categories.

Iteration	Relevant	Irrelevant
Documents Retrieved	a	b
Documents not Retrieved	c	d

Table 2. Documents possible sets based on a query in *IR*

P (*document* | *class*) represents the probability of document given class, and documents can be modeled as sets of words, thus the *p* (*docment* | *class*) can be written as:

$$P\ (document \mid class) = \prod p\ (wordi \mid class) \qquad (2)$$

Therefore:

$$P\ (class||\ document) = \prod p\ (class)\ p\ (wordi\ |\ class) \qquad (3)$$

Category Name	NB			K-NN			NN		
EvaluationMeasures	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
Medical	0.95	0.95	0.95	0.42	1	0.59	0.92	0.93	0.92
Information Technology	0.97	0.78	0.76	0.33	0.57	0.42	0.83	0.83	0.81
Sport	0.92	0.95	0.92	0.42	1	0.59	0.92	1	0.96
Entertainment	0.88	0.91	0.91	0.25	0.6	0.35	0.77	0.92	0.83
Local News	0.77	0.71	0.74	0.08	1	0.15	0.52	0.72	0.61
Economics	0.78	0.84	0.81	0.75	0.17	0.28	0.67	0.77	0.71
International News	0.93	0.83	0.87	0.33	0.34	0.38	0.82	0.92	0.86
Average	0.85	0.85	0.85	0.37	0.37	0.37	0.78	0.87	0.81

Table 3. Results F1, Recall and Precision of Arabic Text Categorization

Where: *P* (*wordi* | *class*): The probability that the *i-th* word of a given document occurs in a document from class *C*, and this can be computed as follows:

$$P\ (wordi\ |\ class) = (Tct + \lambda)\ /\ (Nc + \lambda V) \qquad (4)$$

Where

Tct : The number of times the word occurs in that category *C*

Nc : The number of words in category *C*

V : The size of the vocabulary table

λ: The positive constant, usually 1 or 0.5 to avoid zero probability.

KNN is a classification algorithm where objects are classified by voting several labeled training examples with

$$Sim\ (X, D_j) = \frac{\sum t_i \in\ (X \cap D_j)\ x_i \times d_{ij}}{\| X \|_2 \times \| D_j \|_2}$$

Modern Neural Networks are descendants of the perceptron model and the least mean square (LMS) learning systems of the 1950s and 1960s. The perceptron model and its training procedure were presented for the first time by Rosemblatt (1962), and the current version of the LMS is due to Widrow and Hoff (1960). The perceptron is a device that decides whether an input pattern belongs to one of two classes. The mathematical model of the perceptron corresponds to a linear discriminant and can be written as:

$$\sum_{1} w_i\, I_i + \theta$$

their smallest distance from each object. KNN was initially applied to classification of news articles by Massand et al, in 1992 [12]. Yang compared 12 approaches to text categorization with each other, and judged that KNN is one of recommendable approaches, in 1999 [20]. KNN is evaluated as a simple and competitive algorithm with Support Vector Machine for implementing text categorization systems by Sebastiani in 2002 [18]. Its disadvantage is that KNN costs very much time for classifying objects, given a large number of training examples because it should select some of them by computing the distance of each test object with all of the training examples.

4. Experimental Result and Discussion

Three evaluation measures (Recall, Precision, and *F*1) as the bases of our comparison, where *F*1 is computed based on the following equation:

$$F1 = \frac{2 * Precision * Recall}{Recall + Precision} \tag{5}$$

Precision and recall are widely used evaluation measures in *IR* and *ML*, where according to Table 2,

$$Precision = \frac{a}{(a + b)} \tag{6}$$

$$Recall = \frac{a}{(a + c)} \tag{7}$$

To explain precision and recall, let's say someone has 5 blue and 7 red tickets in a set and he submitted a query to retrieve the blue ones. If he retrieves 6 tickets where 4 of them are blue and 2 that are red, it means that he got 4 out of 5 blue (1 false negative) and 2 red (2 false positives). Based on these results, precision = 4 / 6 (4 blue out of 6 retrieved tickets), and recall = 4 / 5 (4 blue out of 5 in the initial set).

Table 3 gives the *F*1, Recall, and Precision results generated by the three categorizers (*NB*, *k-NN* and *NN*) against *SNP* data sets where in each data set using ten-fold cross-validation. Cross validation is a known evaluation method in data mining, where the training data is divided randomly into *n* blocks, each block is held out once, and the classifier is trained on the remaining *n*–1 blocks; then its error rate is evaluated on the holdout block. Therefore, the learning procedure is executed n times on slightly different training data sets.

All the experiments were conducted using The Weka open source software [23]. After analysing Table 3, we found that the NB categorizer outperformed on six data sets with regards to *F*1 results. Precision results obtain that the NB outperformed the other two on all data.

Recall results demonstrate that the *NN* outperformed *NB* and *k*-NN on five data sets, *k*-NN is in the lead in 3 data sets, and NB comes last in one data set. The average of the three measures obtained against eight Arabic data sets indicated that NB algorithm is dominant.

5. Conclusions and Future Works

In this paper we discussed the problem of automatically classifying Arabic text documents. We used the NB algorithm which is based on probabilistic framework. We also used *K-NN* algorithm along with NN to handle our classification problem.

The average of the three measures obtained against SNP Arabic data sets indicated that NB categorizer outperformed both *k-NN* and NN algorithms with regard to *F*1, Recall and Precision measures.

In near future, we intend to propose a new approach based for enhancing the training of the classifier for text categorization problem.

References

[1] Benkhalifa, M., Mouradi, A., Bouyakhf, H. (2001). Integrating WordNet knowledge to supplement training data in semi-supervised agglomerative hierarchical clustering for text categorization, *Int. J. Intel Syst* (16:8), p.929-947.

[2] Guo, G., Wang, H., Bell, D., Bi, Y., Greer, K. (2004). An kNN Model-based Approach and its Application in Text Categorization, *In*: proceedings of 5th *I*nternational Conference on Intelligent Text Processing and Computational Linguistic, CICLing, LNCS 2945, Springer-Verlag, p.559-570.

[3] Hadi, W., Thabtah, F., ALHawari, S., Ababneh, J. (2008b). Naive Bayesian and K-Nearest Neighbour to Categorize Arabic Text Data. *In*: Proceedings of the European Simulation and Modelling Conference. Le Havre, France, p. 196-200.

[4] El-Halees, A. (2006). Mining Arabic Association Rules for Text Classification *In*: the proceedings of the first international conference on Mathematical Sciences, *Al-Azhar University of Gaza, Palestine*, p.15 -17.

[5] El-Halees, A. (2007). Arabic Text Classification Using Maximum Entropy The Islamic University, *Journal of Series of Natural Studies and Engineering* (15, 1), p.157-167.

[6] Hammo, B., Abu-Salem, H., Lytinen, S., Evens, M. (2002). QARAB: A Question Answering System to Support the Arabic Language. *Workshop on Computational Approaches to Semitic Languages*. ACL, Philadelphia, PA, July. p. 55-65.

[7] Al-Harbi, S., Almuhareb, Al-Thubaity, A., Khorsheed, M. S., Al-Rajeh, A. Automatic Arabic Text Classification. JADT: 9es *Journées Internationales d'Analyse Statistique des Données Textuelles*. p. 77-83.

[8] Joachims T. (1999). Transductive Inference for Text Classification using Support Vector Machines. *In*: Proceedings of the International Conference on Machine Learning (ICML), p. 200-209.

[9] Joachims, T. (1998). Text Categorization with Support Vector Machines: Learning with Many Relevant Features, *In*: Proceedings of the European Conference on Machine Learning (*ECML*), p.173-142, Berlin.

[10] El-Kourdi, M., Bensaid, A., Rachidi, T. (2004). Automatic Arabic Document Categorization Based on the Naïve Bayes Algorithm, 20*th* *International Conference on Computational Linguistics*, 2004, Geneva.

[11] Laila K. Arabic Text Classification Using N-Gram Frequency Statistics A Comparative Study, *DMIN*, p.78-82.

[12] Massand, B., Linoff, G., Waltz, D. (1992). Classifying News Stories using Memory based Reasoning, *In*: Proceedings [13] of 15th ACM International Conference on Research and Development in Information Retrieval, p. 59-65.

[14] Mesleh, A. A. (2007). Chi Square Feature Extraction Based Svms Arabic Language Text Categorization System, *Journal of Computer Science* (3:6), p. 430-435

[15] Moulinier, I., Raskinis, G., Ganascia, J. (1996). Text categorization: a symbolic approach, *In*: Proceedings of the Fifth Annual Symposium on Document Analysis and Information Retrieval.

[16] Quinlan, J. (1993). C4.5: Programs for machine learning. San Mateo, CA: Morgan Kaufmann.

[17] Sakhr software company's website: www.sakhrsoft.com, (2004) .

[18] Sebastiani, F. (2002). Machine Learning in Automated Text Categorization, *ACM Computing Survey*, 34 (1) 1-47.

[19] Samir, A., Ata, W., Darwish, N. (2005). A New Technique for Automatic Text Categorization for Arabic Documents, 5*th* *IBIMA Conference* (*The internet & information technology in modern organizations*), Cairo, Egypt.

[20] Sawaf, H., Zaplo, J., Ney, H. (2001). Statistical Classification Methods for Arabic News Articles. *Arabic Natural Language Processing*, Workshop on the ACL. Toulouse, France.

[21] Yang, Y. (1999). An evaluation of statistical approaches to text categorization, *Information Retrieval*, 1 (1-2) 67-88.

[22] Sebastiani, F. (1999). A Tutorial on Automated Text Categorization, *In*: Proceedings of the ASAI-99, 1st Argentinian Symposium on Artificial Intelligence, p. 7-35.

[23] Thabtah, F., Eljinini, M., Zamzeer, M., Hadi, W. (2009). Naïve Bayesian based on Chi Square to Categorize Arabic Data. *In*: proceedings of The 11th International Business Information Management Association Conference (IBIMA) Conference on *Innovation and Knowledge Management in Twin Track Economies*, Cairo, Egypt 4 - 6 January. (p. 930-935).

[24] Thabtah, F., Hadi, W., Al-shammare, G. (2008). VSMs with K-Nearest Neighbour to Categorise Arabic Text Data. *In*: The World Congress on Engineering and Computer Science. p.778-781, 22-44 October. San Francisco, USA.

[25] Van Rijsbergan, C. (1979). Information retrieval Buttersmiths, London, 2nd Edition.

[26] Vapnik, V. (1995). The Nature of Statistical Learning Theory, chapter 5. Springer-Verlag, New York.

[27] WEKA. (2001). Data Mining Software in Java: http://www.cs.waikato.ac.nz/ml/weka.

[28] Wiener, E., Pedersen, J. O.,Weigend, A. S. A neural network approach to topic spotting. *In*: Proceedings of the Fourth Annual Symposium on Document Analysis and Information Retrieval (SDAIR), p. 317-332, Las Vegas, Nevada.

[29] Lam, W., Ho, C. Y. (1998). Using a Generalized Instance Set for Automatic Text Categorization, SIGIR'98, p. 81-89.

[30] Elmougy, S., Ghoneim, A., Hamza, T. (2010). Naive Bayes Classifier based Arabic document categorization, The 7th International Conference on Informatics and Systems (INFOS).

[31] Bawaneh, M. J., Alkoffash, M. S., Al Rabea, A. I. (2008). Arabic Text Classification using K-NN and Naïve Bayes. *Journal of Computer Science*, 4 (7) 600-605.

[32] Duwairi, R. (2007). Arabic Text Categorization. *The International Arab Journal of Information Technology*, 4 (2), April.

[33] Al-Shammari, E. (2009). A Novel Algorithm for Normalizing Noisy Arabic Text. *World Congress on Computer Science and Information Engineering*.