Journal of King Saud University – Computer and Information Sciences (2015) **27**, 437–449

King Saud University

Journal of King Saud University – Computer and Information Sciences

www.ksu.edu.sa
www.sciencedirect.com

Arabic text classification using Polynomial Networks

Mayy M. Al-Tahrawi [a,*], **Sumaya N. Al-Khatib** [b]

[a] *Computer Science Department, Faculty of Information Technology, Al-Ahliyya Amman University, Amman, Jordan*
[b] *Software Engineering Department, Faculty of Information Technology, Al-Ahliyya Amman University, Amman, Jordan*

Received 7 May 2014; revised 6 November 2014; accepted 14 February 2015
Available online 10 September 2015

KEYWORDS

Polynomial Networks;
Arabic text classification;
Arabic document categorization

Abstract In this paper, an Arabic statistical learning-based text classification system has been developed using Polynomial Neural Networks. Polynomial Networks have been recently applied to English text classification, but they were never used for Arabic text classification. In this research, we investigate the performance of Polynomial Networks in classifying Arabic texts. Experiments are conducted on a widely used Arabic dataset in text classification: Al-Jazeera News dataset. We chose this dataset to enable direct comparisons of the performance of Polynomial Networks classifier versus other well-known classifiers on this dataset in the literature of Arabic text classification. Results of experiments show that Polynomial Networks classifier is a competitive algorithm to the state-of-the-art ones in the field of Arabic text classification.

1. Introduction

With the rapid growth in the availability and use of natural language electronic texts, automatic Text Classification (TC) becomes an important technique for understanding and organizing such texts. TC automatically assigns an unseen document to one or more pre-defined classes based on the document content. It is used in many areas, such as digital libraries, spam filtering, online news, word sense disambiguation, information retrieval and topical crawling. Automatic TC is needed heavily due to the huge amount of text on the web which cannot be classified manually by human experts due to cost and time considerations.

The bulk of the efforts on TC work have been devoted to automatic classification of English and Latin texts (Yang and Liu, 1999; Fang et al., 2001; Sebastiani, 2002; Joachims, 2002; Crammer and Singer, 2003; Lewis et al., 2004). Researchers paid little interest for investigating TC approaches in classifying Arabic texts despite the fact that the Arabic language is one of the seven official languages of the United Nations with more than 400 million native speakers. Furthermore, a large percentage of these native Arabic speaking users cannot read English.

The limited research work in Arabic TC can be attributed to many reasons: the complex morphology of the Arabic

[*] Corresponding author at: P.O. Box: 348, 19374 Amman, Jordan. Mobile: +962 79 5414927, +962 78 6702047.
E-mail addresses: mtahrawi@ammanu.edu.jo, mayy.tahrawi@gmail.com, tahrawi_mayy@yahoo.com (M.M. Al-Tahrawi), sumayakh@ammanu.edu.jo (S.N. Al-Khatib).
Peer review under responsibility of King Saud University.

Production and hosting by Elsevier

http://dx.doi.org/10.1016/j.jksuci.2015.02.003
1319-1578 © 2015 The Authors. Production and hosting by Elsevier B.V. on behalf of King Saud University.

language, the wide spread of synonyms in the Arabic Language, the high inflectional and derivational nature of the Arabic language, the lack of availability of publicly free accessible Arabic Corpora and finally the lack of standard Arabic morphological analysis tools. In fact, all researchers on Arabic TC have concluded that building Arabic text classifiers is a challenging task (Khreisat, 2006; Harrag and El-Qawasmeh, 2009; El-Halees, 2007; Duwairi, 2007).

Nevertheless, the need and interest in classifying Arabic language texts have grown lately, due to many reasons: Arabic language is very rich with documents, there are tens of millions of Arab Internet users and a large percentage of these users cannot read English pages. Add to this, the Arabic internet content has grown rapidly in the last years, exceeding 3% of the whole internet content and is ranked the eighth in the whole internet content (http://www.InternetWorldStats.com). This continuously-growing content needs to be exchanged and thus automatically and efficiently classified.

One Arabic automatic categorizer, "Sakhr's categorizer" (Sakhr, 2004) has been reported to have been put under operational use to classify Arabic documents. No technical documentation or specification concerning this Arabic categorizer is available.

Recently, researchers started to investigate the performance of some well-known English TC algorithms in classifying Arabic text documents. Examples include the Naïve Bayes algorithm (NB) (Yahyaoui, 2001; El-Kourdi et al., 2004; Duwairi, 2007; El-Halees, 2008; Kanaan et al., 2009; Al-Saleem, 2010, 2011; Chantar and Corne, 2011; Khorsheed and Al-Thubaity, 2013; Belkebir and Guessoum, 2013; Sharef et al., 2014), Support Vector Machines (SVM) (Mesleh, 2007; Al-Harbi et al., 2008; El-Halees, 2008; Said et al., 2009; Al-Saleem, 2010, 2011; Chantar and Corne, 2011; Khorsheed and Al-Thubaity, 2013), k-Nearest Neighbor (kNN) (Al-Shalabi et al., 2006; Duwairi, 2007; El-Halees, 2008; Kanaan et al., 2009; Khorsheed and Al-Thubaity, 2013; Ababneh et al., 2014) and decision tree (Al-Harbi et al., 2008; El-Halees, 2008; Harrag et al., 2009; Chantar and Corne, 2011; Khorsheed and Al-Thubaity, 2013) besides others (Sawaf et al., 2001; Duwairi, 2005, 2007; Khreisat, 2006; Ghwanmeh, 2007; El-Halees, 2007, 2008; Al-Harbi et al., 2008; Kanaan et al., 2009; Khorsheed and Al-Thubaity, 2013; Belkebir and Guessoum, 2013; Fodil et al., 2014).

Polynomial neural Networks (PNs) are a supervised machine learning algorithm that draws on traditional mathematical methods and evolutionary programing concepts to evolve a network of polynomial functions capable of approximating any continuous multivariate function from a collection of input–output data. They differ from artificial neural networks in that they have neither biological inspiration nor interpretation. PNs were first used for TC in 2008 (AL-Tahrawi and Abu Zitar, 2008).They were not used earlier in TC, as the requirements of PN techniques grow exponentially with the model complexity and the number of features used. Nevertheless, (AL-Tahrawi and Abu Zitar, 2008; AL-Tahrawi, 2014, 2015) have proved that PNs are competitive English text classifiers to the state-of-the-art ones in this field, including SVM, KNN, NB, Logistic Regression (LR) and Radial Basis Function Networks (RBF).

In this research, PNs are investigated in classifying Arabic text documents for the first time in the literature of Arabic TC. The rest of the paper is organized as follows: Related work on Arabic TC is presented in Section 2, PN classification algorithm is presented in detail in Section 3, the Dataset and data preprocessing are presented in Section 4, Experiments, Results and Analysis of results are presented in Section 5 and finally Conclusions are presented in Section 6.

2. Related work

Although a lot of works have studied classification of English and Latin texts very early, only few works have studied the classification of Arabic texts in the last decade. Such studies address the problem of TC using different Datasets, Data Pre-processing methods, Feature Selection methods, Classification methods, as well as different metrics to evaluate the performance of these classifiers.

2.1. DataSets

Unlike the case in English TC, there are no free benchmark datasets available for the researchers in Arabic TC; As a result, many researchers depend on collecting their own in-house data sets (Khreisat, 2006; Duwairi, 2007;Kanaan et al., 2009; Al-Saleem, 2010; Fodil et al., 2014), which are gathered from different resources, like News Channels and Websites. Some datasets are made available for free use by researchers, like Alj-News which was used by El-Kourdi et al. (2004), El-Halees (2007, 2008), Mesleh (2007), Said et al. (2009), Chantar and Corne (2011), Open Source Arabic Corpora (OSAC) (Belkebir and Guessoum, 2013) and Saudi Newspapers (SNP) (Al-Harbi et al., 2008; Al-Saleem, 2011; Ababneh et al., 2014).

The number of (documents, classes) in these corpora vary from just (175,5) (Fodil et al., 2014) to (33 K, 34) (Sawaf et al., 2001). Some researchers do not clarify this important piece of information regarding their corpora (Khreisat, 2006; El-Halees, 2007, 2008).

Regarding the dataset split into training/testing, there is no agreement upon the split of the dataset into training/testing parts, even when using the same data set. Furthermore, some researchers do not even mention the training/testing split of the dataset they use (El-Halees, 2007; Fodil et al., 2014).

2.2. Text pre-processing in the literature of Arabic TC

Data pre-processing is considered an important part in building text classifiers. The main advantage of applying data preprocessing on the text documents is reducing the number of features (terms) in the dataset, as well as enhancing classifiers performance in terms of resource requirements and classification accuracy. Many researchers in the field of Arabic TC apply a set of text pre-processing steps on the texts before classification, like the exclusion of stop words, punctuation marks, diacritics, non letters and vowels, as well as normalization of some letters like al hamza (El-Kourdi et al., 2004; Khreisat, 2006; Duwairi, 2007; El-Halees, 2007, 2008; Mesleh, 2007; Al-Harbi et al., 2008; Kanaan et al., 2009; Said et al., 2009; Al-Saleem, 2010, 2011; Chantar and Corne, 2011; Khorsheed and Al-Thubaity, 2013; Belkebir and Guessoum, 2013; Sharef et al., 2014; Fodil et al., 2014; Ababneh et al., 2014). Some researchers also removed infrequent words (Mesleh, 2007; Chantar and Corne, 2011) or words less than a certain length (Sharef et al., 2014).

Another major pre-processing step in TC is either Stemming or Root Extraction. This step aims to reduce words to their stem or root, resulting in reducing the number of the terms the classifier needs to work with. This results in reducing memory and processing requirements of the classifier system. Researchers in Arabic TC used three different types of stemming in building Arabic TC systems: Stemming (El-Halees, 2007, 2008; Said et al., 2009; Fodil et al., 2014), Light Stemming (Said et al., 2009; Belkebir and Guessoum, 2013; Sharef et al., 2014) and/or Root Extraction (Duwairi, 2007; Said et al., 2009; Belkebir and Guessoum, 2013). Yet, some researchers did not apply any kind of stemming or root extraction in building their classifiers (Mesleh, 2007; Chantar and Corne, 2011).

On the other hand, some researchers did not perform any type of pre-processing on the dataset (Sawaf et al., 2001).

2.3. Feature weighting and selection

Feature Selection (FS) is widely used in TC, as most classifiers cannot work with the huge number of terms in the corpus. Add to this, the effect of using all terms (features) in building classifier on the classifier accuracy was always a great debate; many researchers believe that using all corpus terms adds both noise and processing requirements to the classifiers, without enhancing classification accuracy, while others found FS harmful to TC (Khreisat, 2006). Using FS, the discriminating power of each term is computed, and only the top-scoring ones are used to build the classifier.

Several FS methods are used in the field of Arabic TC research, like Cross Validation (El-Kourdi et al., 2004), Chi Square (CHI) (Mesleh, 2007; Al-Harbi et al., 2008; Khorsheed and Al-Thubaity, 2013; Belkebir and Guessoum, 2013; Sharef et al., 2014), Information Gain(IG) (El-Halees, 2008; Said et al., 2009; Khorsheed and Al-Thubaity, 2013), Document Frequency (DF) (Said et al., 2009; Khorsheed and Al-Thubaity, 2013), Mutual Information (MI) (Said et al., 2009), Correlation Coefficient (CC) (Said et al., 2009), Binary Particle Swarm Optimization-K-Nearest-Neighbor (BPSO-KNN) (Chantar and Corne, 2011), Semi-Automatic Categorization Method (SACM) and Automatic Categorization Method (ACM) (Fodil et al., 2014). On the other hand, (Sawaf et al., 2001) selected features randomly and (Khreisat, 2006) didn't use any FS.

After deciding on the features to be selected for building the classifier, the features will be represented in the classification system using one of the various presentations or weights used in the literature of TC. Common examples include Term Frequency. Inverse Document Frequency (TF.IDF) (El-Kourdi et al., 2004; Mesleh, 2007; Kanaan et al., 2009; Chantar and Corne, 2011; Belkebir and Guessoum, 2013; Fodil et al., 2014), Term Frequency (TF) (Khreisat, 2006; Kanaan et al., 2009; Khorsheed and Al-Thubaity, 2013; Sharef et al., 2014; Fodil et al., 2014), Document Frequency (DF) (Khorsheed and Al-Thubaity, 2013), Weighted IDF (Kanaan et al., 2009), Normalized Frequency (Sawaf et al., 2001; El-Halees, 2008), Boolean (Al-Harbi et al., 2008; Khorsheed and Al-Thubaity, 2013), Binary (Al-Harbi et al., 2008; Khorsheed and Al-Thubaity, 2013) and other FS methods like Cosine coefficient, Dice coefficient and Jaccard coefficient (Ababneh et al., 2014).

2.4. Classification algorithms

Several Classification Algorithms were experimented in the literature of Arabic TC. Some well-known algorithms in English TC were successful in Arabic TC, like Support Vector Machines (SVM) (Mesleh, 2007; Al-Harbi et al., 2008; El-Halees, 2008; Said et al., 2009; Al-Saleem, 2010, 2011; Chantar and Corne, 2011; Khorsheed and Al-Thubaity, 2013), Naïve Bayes (NB) (El-Kourdi et al., 2004; Duwairi, 2007; El-Halees, 2008; Kanaan et al., 2009; Al-Saleem, 2010, 2011; Chantar and Corne, 2011; Khorsheed and Al-Thubaity, 2013; Belkebir and Guessoum, 2013; Sharef et al., 2014), K-Nearest_Neighbor (kNN) (Duwairi, 2007; El-Halees, 2008; Kanaan et al., 2009; Khorsheed and Al-Thubaity, 2013; Ababneh et al., 2014), Maximum Entropy (Sawaf et al., 2001; El-Halees, 2007, 2008), Artificial Neural Network (ANN) (El-Halees, 2008; Belkebir and Guessoum, 2013; Khorsheed and Al-Thubaity, 2013), Decision Tree (DT) (Al-Harbi et al., 2008; El-Halees, 2008; Chantar and Corne, 2011; Khorsheed and Al-Thubaity, 2013) and the Rocchio feedback algorithm (Kanaan et al., 2009).

2.5. Performance evaluation

After building a classifier, its performance has to be evaluated using some formal measure like Accuracy (El-Kourdi et al., 2004; Al-Harbi et al., 2008; Khorsheed and Al-Thubaity, 2013; Belkebir and Guessoum, 2013), Precision (Sawaf et al., 2001; Khreisat, 2006; Duwairi, 2007; El-Halees, 2007, 2008; Kanaan et al., 2009; Al-Saleem, 2010, 2011; Chantar and Corne, 2011; Ababneh et al., 2014), Recall (Sawaf et al., 2001; Khreisat, 2006; Duwairi, 2007; El-Halees, 2007, 2008; Kanaan et al., 2009; Al-Saleem, 2010, 2011; Chantar and Corne, 2011; Ababneh et al., 2014), F-measure (Sawaf et al., 2001; El-Halees, 2007, 2008; Mesleh, 2007; Said et al., 2009; Al-Saleem, 2010, 2011; Chantar and Corne, 2011; Sharef et al., 2014; Ababneh et al., 2014), fallout (Duwairi, 2007) and error rate (Duwairi, 2007).

The formulae for computing some of these measures are provided in Section 5.2.

Table 1 summarizes a number of these studies. The table presents, for each research work, the Corpus used, the split of the corpus into training and testing parts, the Data Pre-Processing applied to the corpus documents, Feature Weighting methods and Selection criteria, the classification algorithm used and finally the performance achieved in each research. Papers are presented in chronological order in the table.

As is clear from the various research works presented in Table 1, there is no agreement on the dataset, its size, number of classes, or even on the preprocessing steps applied on the documents. This makes direct and thus fair comparisons very difficult.

3. Polynomial Networks (PNs)

Polynomial neural Network (PN) classifiers have been known in the literature for many years (Fukunaga, 1990; Campbell et al., 2001; Assaleh and Al-Rousan, 2005; Liu, 2006). Recently, PNs have proved to be competitive to the top performers in the field of English TC of the two benchmark datasets: Reuters and 20Newsgroups, using only 0.25–0.5%

Table 1 A summary of a related research work in Arabic TC.

Reference	Corpus				Preprocessing	Training/ testing split	Feature weight	Feature selection	Classification algorithm	Performance
	DataSet	Genre	# Docs	# Classes						
Sawaf et al. (2001)	Arabic NEWSWIRE 1994	News	33 K	10, 34	No	0.80/0.20	Normalized frequency	Random	Maximum entropy	Precision: 50.0 Recall: 89.5 F-measure: 62.7
El-Kourdi et al. (2004)	Al-Jazeera News	News	1500	5	Exclusion of stop words, stripping vowels, root extraction	0.333/ 0.667 0.50/0.50 0.6670.333	TF.IDF	Cross validation	NB	Average accuracy: 68.78% Best accuracy: 92.8%.
Khreisat (2006)	Jordanian newspapers (Al-Arab, Al-Ghad, Al-Ra'I, Ad-Dostor)	News	N.A	4	Removal of punctuation marks, stop words, diacritics, and non letters. Replacing initial أ, آإ with ا. Replacing final ى followed by ء with ئ.	0.40/0.60	TF	No	Manhattan distance, Dice measure	Macro Average Precision and Recall: Manhattan measure: (0.665, 0.56). Dice measure: (0.8875, 0.83)
Duwairi (2007)	In-house collected	News	1000	10	Removal of punctuation marks, formatting tags, prepositions, pronouns, conjunction and auxiliary verbs. Root extraction	0.50/0.50	N.A.	N.A	KNN, NB, distance-based	NB recorded the best accuracy with the highest Precision/Class:1 and the lowest Precision/class: 67 Distance-based comes last with Micro average Precision, Recall, fallout, and error rate: (74.0,62.8,4.1,7.4)
El-Halees (2007)	Aljazeera Arabic News www.elaph.net, www.palestine-info.info and www.islamonlone.net	News	N.A.	6	Removal of punctuations and non-letters Converting آ to ا Replacing ى by ي and ة by ه Removal of stop words Stemming	N.A	N.A.	N.A	Maximum Entropy	Recall: 80.48 Precision: 80.34 F-measure: 80.41
Mesleh (2007)	Al-Jazeera Al-Nahar Al-Hayat Al-Ahram Al-Dostor	News	1445	9	Removal of digits and punctuation marks. Filtering all non-Arabic words. Exclusion of stop words, diacritics, non-letters and prepositions. Normalization of hamza. Removal of Infrequent terms.	0.667/ 0.333	TF.IDF	CHI	SVM	Macro-average F 88.11
Al-Harbi et al. (2008)	Saudi Press Agency SPA SNP WEB Sites Writers Discussion Forums Islamic Topics Arabic Poems	Various	17,658	7	Exclusion of stop words	0.70/0.30	Binary	CHI	SVM and C5.0	Average accuracy: -SVM: 68.65% -C5.0: 78.42%

El-Halees (2008)	Aljazeera Channel website	News	N.A	6	Exclusion of stop words, punctuation marks, diacritics, and non-letters Converting آ to ا Replacing ى by ي and ة by ه Stemming	10-fold cross validation	Normalized frequency	IG	Maximum Entropy, NB, KNN, DT, SVM, ANN	Precision, Recall, and f-measure. NB (without FS) outperformed all algorithms, F(91.81) Precision, Recall, and f-measure. SVM (with I.G) outperformed all algorithms, F(88.33)
Kanaan et al. (2009)	Newspapers websites	News	1445	9	Exclusion of stop words. Removal of punctuation marks, diacritics and non-letters.	4-fold cross validation	TF TF.iDF Weighted IDF	N.A.	KNN, NB and Rocchio	NB outperformed others using Precision and Recall
Said et al. (2009)	Alj-News Arabic Dataset Alj-Magazine Arabic Dataset	News	1500 4470	5 N.A.	(1) Stemming using three Stemmers: RDI MORPHO3 Sebawai Root Extractor (SR) Light Stemmer (AS) (2) Removal of Stop words.	1200/300 Cross-validation	N. A.	DF IG MI CC	SVM	Mico-F1 results are provided as Figures in their research. AS with MI or IG recorded best performance
Al-Saleem (2010)	Newspapers websites	News	5121	7	Exclusion of stop words, punctuation marks, diacritics, and non-letters. Normalization	10-fold cross validation	N.A.	N.A	CBA, NB and SVM	CBA outperformed macro average Precision, Recall and F-measure. (80.5, 80.7, 80.4)

Chantar and Corne (2011)	Akhbar-Alkhaleej online newspaper
	Alwatan online newspaper
	Al-jazeera-News
		News
		News
		News
			1708
			1173
			1500
				4
				4
				5
					Removal of hyphens, punctuation marks, numbers, digits, non-Arabic letters and diacritics.
					Removal of stop words and rare words that occur less than five times in the dataset
					No stemming.
					No normalization of some Arabic letters.
						1365/343
						821/352
						1200/300
							TF.IDF	BPSO-KNN	NB,
									J48
									SVM
										Best performance was on Alj_News:
										(Precision, Recall, F-Measure)
										SVM(0.937, 0.93, 0.931)
										NB(0.858, 0.843, 0.846)
										J48(0.747, 0.723, 0.729)

Al-Saleem (2011)	SNP	News	5121	7	Removal of digits and punctuations
					Normalization of Hamza.
					Filtering all the non Arabic texts.
					Removal of stop words.
						10-fold cross-validation.	N. A.	N. A.	SVM
									NB
										Average Precision, Recall
										F-measure
										SVM(0.779 0.778 0.778)
										NB(0.741 0.74 0.74)

Khorsheed and Al-Thubaity (2013)	King Abdulaziz city for Science and Technology corpus	Saudi press agency Saudi newspapers Websites Writers	17,658	10	Removal of numbers, punctuations, kashida and stop words. Normalization of the Hamza.	0.70/0. 30	TF DF Binary	DF IG CHI	kNN NB SVM C4.5 ANN	Best accuracy: NB: 72.69

(continued on next page)

Table 1 (*continued*)

Reference	Corpus				Preprocessing	Training/ testing split	Feature weight	Feature selection	Classification algorithm	Performance
	DataSet	Genre	# Docs	# Classes						
		Forums Islamic topics Arabic poems								
Belkebir and Guessoum (2013)	OSAC	News	1000	10	Removal of digits, Latin alphabet, isolated letters, punctuation marks, stop words and diacritics. Normalization of HAMZA. - Root-based stemming - Light stemming.	0.70/0. 30	TF.IDF	CHI	ANN SVM BSO-CHI-SVM	Best accuracy: BSO-CHI-SVM (95.67%)
Sharef et al. (2014)	N.A.	N.A.	3172	4	Removal of digits, punctuation marks, non-Arabic words, stop words. Normalizing the aleph and hamza letters. Light stemming Removing all the words with length less than three.	Random	TF	CHI	Frequency Ratio Accumulation Method (FRAM) NB Multi-variant Bernoulli Naïve Bayes (MNB) Multinomial Naïve Bayes (MBNB)	FRAM achieved the best macro-average F1: (95.1%) using Bag-Of-Word (BOW) (93.6%) using 3-gram character level representation
Fodil et al. (2014)	ADTC1 (Arabic Dataset for Theme Classification, subset 1) ADTC2 (Arabic Dataset for Theme Classification, subset 2).).	News books	175	5	Removal of punctuation marks, diacritics, numbers, non Arabic letters, and kashida except in the term Allah. Normalizing some writing forms that include “ ء” “ ى”, “ ة ” to “ ا”, “ ي ” and ه“ Removal of stop words Stemming	N.A.	TF TF.IDF	SACM ACM	The Cumulative Thematic Probability (CTP)	Global recognition score measures the percentage of documents that are correctly assigned in each category: using TF. IDF 95% using TF 88%
Ababneh et al. (2014)	SNP	News	5121	7	Normalization of hamza Filtering all the non-Arabic texts Removal of stop words.	0.70/0.30	Cosine coefficient, Dice coefficient and Jacaard coefficient	N.A.	kNN	Cosine outperformed Dice and Jaccard With the best class-level results: Precision: 0.917 Recall: 0.979 F1: 0.947

N.A. Not Available.

of the corpora terms (AL-Tahrawi and Abu Zitar, 2008; AL-Tahrawi, 2013, 2014, 2015).

Several Neural Network approaches may be used to classify different types of data. In this research, we use the Polynomial Neural Networks algorithm proposed by Campbell et al. (2001) to classify Arabic text documents. The proposed algorithm uses discriminative training with a mean-squared error criterion. Details of the algorithm and its application in TC are explained in the following subsections.

3.1. The architecture of PNs

The representation of the PN model adopted in this research consists of two layers. In the first layer (the input layer), the set of inputs (features) $x(x_1, x_2, \ldots, x_N)$, where N is the number of input features, are used to form a set of monomial basis functions $p(x)$ of the required order or degree K. One basis function $p(x)$ is formed for each observation.

The elements of $p(x)$ for a polynomial of degree K are monomials of the form (Campbell et al., 2001):

$$\prod_{j=1}^{N} x_j^{k_j}, \text{ where } k_j \geqslant 0 \text{ and } 0 \leqslant \sum_{j=1}^{N} k_j \leqslant K \quad (1)$$

For example, if an input vector x contains the two features x_1 and x_2, the second order polynomial network basis function $p(x)$ will look as follows:

$$p(x) = [1 \ x_1 \ x_2 \ x_1^2 \ x_1x_2 \ x_2^2]^t \quad (2)$$

Polynomials of degree 2 were used in this research, as this degree recorded the best performance results in our experiments.

Then, the second layer of the PN combines all the outputs of the first layer (the basis functions) to compute scores $w^t\,p(x)$, where w is the classification model. A score $w_j^t\,p(x_i)$ is produced for each input vector $\mathrm{x_i}$ and each class j. Then, the final output is computed by averaging the total score over all feature vectors (Campbell et al., 2001):

$$s_j = \frac{1}{M}\sum_{i=1}^{M} w^t p(x_i) \quad (3)$$

where M is the number of feature vectors in class j. This final score will be used to recognize and verify new unseen inputs.

That is to say, the data are first expanded into a high dimensional space in the first layer and then linearly separated using the second layer.

Details of using PNs in TC are explained in Section 3.2.

3.2. The training phase of PN classifiers

A PN is trained to approximate an ideal output using mean squared error as the objective criterion. The polynomial expansion of the ith class term vectors (documents) is denoted by Campbell et al. (2001), AL-Tahrawi and Abu Zitar (2008):

$$M_i = [p(x_{i,1}) \ p(x_{i,2}) \ p(x_{i,3}) \ \ldots \ p(x_{i,Ni})]^t \quad (4)$$

where N_i is the number of training feature vectors for class i, and $p(x_{i,m})$ is the basis function of the mth feature vector for class i. After forming M_i for each class i of the training classes, a global matrix M is obtained for all the classes, by concatenating the individual M_i's computed for each class (Campbell et al., 2001):

$$M = [M_1 \ M_2 \ M_3 \ \ldots \ M_{nc}]^t \quad (5)$$

where nc is the number of training classes. The training problem then reduces to finding an optimum set of weights w (one weight for each class) that minimizes the distance between the ideal outputs (targets) and a linear combination of the polynomial expansion of the training data such that (Campbell et al., 2001; AL-Tahrawi and Abu Zitar, 2008):

$$w_i^{opt} = arg\ \underset{w}{min}||Mw - O_i||_2 \quad (6)$$

where o_i is the ideal output (a column vector which contains N_i ones in the rows where the ith class' data are located in M, and contains zeros otherwise). A class model w_i^{opt} can be obtained in one shot (non-iteratively) by applying the normal equations method (Campbell et al., 2001; AL-Tahrawi and Abu Zitar, 2008):

$$M^t M w_i^{opt} = M^t o_i \quad (7)$$

Finally, w_i^{opt} is computed as follows:

$$w_i^{opt} = (M^t M)^{-1} M^t o_i \quad (8)$$

3.3. Recognition phase of PN classifiers

Classification of a new unseen input consists of two parts: identification and verification. Identification involves finding the best matching class of a new input, given the feature vector of this input. In the verification phase, the claim made in the identification phase is either accepted or rejected. The identification phase proceeds as follows in the PN algorithm: the term vector x of the unseen input is expanded into its polynomial terms $p(x)$ in a manner similar to what was done with the training inputs in the training phase (Eq. (1)). Then, the new unseen input is assigned to the class c such that (Campbell et al., 2001; AL-Tahrawi and Abu Zitar, 2008):

$$c = arg\ \underset{i}{max} w_i^{opt} \cdot p(x) \quad for\ i = 1, 2, \ldots, nc \quad (9)$$

where nc is the number of the predefined classes in the corpus. In verification, a decision to accept or reject a certain classification can be based on using a certain threshold value. In our experiments, we accepted classifications with scores above 0.5, since the output score $w_{i.}\ p(x)$ lies between 0 and 1.

3.4. Text Classification (TC) using PNs

The training phase of TC starts by forming a term vector x for each training document, using the vector space model. Terms are usually represented by their *tf.idf* weights, *binary* weights, *normalized frequencies*, …etc. *Normalized* frequencies were used in our experiments.

Then, the desired order PN basis function is formed for each training document in the corpus as in Eq. (1). PNs of degree 2 are used in the experiments conducted in this research paper, as it recorded the best performance results in our experiments. For example, if the feature vector of a training document is (0.5, 0.2); i.e. the normalized frequencies for term1 and term2 in this document are 0.5 and 0.2 respectively, then the second order PN basis function for this document is

$$p(x) = [1\ 0.5\ 0.2\ 0.25\ 0.1\ 0.04]$$

After forming the basis function for each input (training) document, M_i (the polynomial expansion of class i) is formed as in Eq. (4). Then, the global matrix for all classes M is formed as in Eq. (5). Now, the PN is trained to approximate the ideal output using the mean-squared error criterion as in Eq. (6) and the individual class weights are computed as in Eqs. (7) and (8).

Finally, the classifier is tested on new unseen documents by forming the basis function $p(x)$ for the term vector x of the new document as in Eq. (1) and assigning this document to the nearest class as in Eq. (9).

4. The DataSet

Different Arabic Datasets were used in the little research work in the area of Arabic TC, as no benchmark Arabic dataset exists. We used Aljazeera News Arabic Dataset (Alj-News), available at (Alj-News Dataset) in this research. Alj-News dataset is gathered from Al-jazeera Arabic News Website. The dataset consists of 1500 Arabic news documents distributed evenly among five classes: Art, Economic, Politics, Science and Sport. Each class has 300 documents (240 for training and 60 for testing). We chose this dataset since it was used in several researches in the literature of Arabic TC (Said et al., 2009; Mohamed et al., 2005; Chantar and Corne, 2011), which enables direct comparisons of our results with those achieved in these researches.

The pre-processing steps and the FS applied on this dataset are explained in Sections 4.1 and 4.2.

4.1. Data pre-processing

Arabic language consists of 28 letters (أ ب ت ث ج ح خ د ذ ر ز س ش ص ض ط ظ ع غ ف ق ك ل م ن ه و ي) in addition to the Hamza (ء). Any Arabic letter other than the three long vowels (أ و ي) is a consonant. Several types of diacritics are used in the Arabic language: Fatha, Kasra, Damma, Sukūn, Shadda, Mad (ا) and Tanwin. They act as short vowels which are used to show the correct pronunciation (and sometimes meaning) of the words, since one Arabic word can have different pronunciations (and hence meanings) using different diacritics. For example, the wordسلم has several forms and meanings, such as:

(1) سَلّمَ: say "hello", delivered
(2) سُلّم: ladder
(3) سَلِمَ: saved
(4) سُلِّم: was delivered
(5) سِلْم: safety

The only way to disambiguate the diacritic-less Arabic words is to locate them within the context. Shapes and sounds of these diacritics are listed in Table 2.

The Arabic language differs from the Latin-based alphabets in that it is written from right to left, with different shapes for the same letter according to its position in the word; for example, (هـ, ـهـ, ـه ، ه) are four different shapes for one letter at the beginning, in the middle of, and at the end of a word respectively. Arabic language exhibits two genders: masculine and feminine and three number classes: singular, dual, and plural.

Table 2 Shapes and sounds of Arabic diacritics.

Diacritic	Example	Sound
Fatha	بَ	Ba
Kasra	بِ	Bi
Damma	بُ	Bu
Sukun	بْ	B
Shadda	بّ	Bb
Tanwin	بٍ باً بٌ	Bun, ban, bin
Madd	آ	Aa

The Arabic plurals are divided into two classes: regular and broken. A noun has three cases, the nominative, accusative and genitive. Apparently, Arabic language is very complex and rich, which explains the difficulties in achieving accurate automatic classification results on Arabic documents.

Data pre-processing is a routine part in building TC systems which aims to remove noise and reduce the number of features (terms) in the dataset. This results in reducing processor and memory requirements for building classifiers, as well as getting more accurate classifications. We applied the following pre-processing steps on Alj-News dataset:

(1) Tokenization: converting documents from sequences of characters into sequences of tokens (terms or features) by recognizing delimiters such as white spaces, punctuations, special characters, etc.
(2) Removal of the non-Arabic letters, numbers, diacritics, special characters and punctuations.
(3) Removal of stop words: these include pronouns, conjunctions, and prepositions. We extended the stop word list adopted by Khoja and Garside (1999) to include 478 stop words rather than the list of just 168 stop words adopted by them.
(4) Stemming: is to reduce an inflected or derived word to its stem. The stem needs not be identical to the morphological root of the word; it is usually sufficient that related words map to the same stem, even if this stem is not itself a valid morphological root. The main advantage of this pre-processing step is to reduce the number of terms in a document, and thus reduce computational and storage requirements of TC systems. With the case of the highly derivative Arabic language, in which a large number of words can be formed using one stem, stemming is a valuable tool in reducing complexity of automatic TC.

In this research, we adopt the Stemming algorithm of Khoja (Khoja and Garside, 1999). It is a well-known Aggressive Arabic Stemmer (Root Extractor) which removes the longest suffix and the longest prefix and then matches the remaining word with verbal and noun patterns to extract the root. As an example, Khoja Stemming algorithm would reduce the Arabic words (المدرسة، المدرس، الدرس، الدارس) which mean (the school), (the teacher), (the lesson) and (the learner) respectively, to one root (درس).

The stemmer makes use of several linguistic data files such as a list of all diacritic characters, punctuation characters, definite articles, and stop words. It has been developed in both C + + and Java and is available at (http://zeus.cs.pacificu.edu/shereen/ArabicStemmerCode.zip).

Table 3 Steps of Khoja Stemming algorithm.

Khoja Arabic Root Extractor

1. Format the word by removing any punctuation, diacritics and non-letter characters
2. Ignore stop words
3. Remove the definite article, such as: ال وال بال كال فال.
4. Remove the special prefix (و)
5. Remove and duplicate the last letter, if the last letter is a shadda
6. Replace أ إ آ with ا
7. Remove Prefixes. للـ لـ سـ ف
8. Remove Suffixes, such as: كن هما كما
9. Match the result against a list of Patterns, such as: فاعل افعل تفعيل فعال
10. Replace all occurrences of Hamza, such as: ؤ ئ ء with ا
11. Two letter roots are checked to see if they should contain a double character; if so, the character is added to the root

The authors in Sawalha and Atwell (2008) evaluated Arabic Language Morphological Analyzers and Stemmers and reported that Khoja stemmer achieved the highest accuracy in their experiments. The stemmer has also been used as part of an Information Retrieval system developed at the University of Massachusetts for the TREC-10 cross-lingual track in 2001. The authors in Larkey and Connell (2001) reported that although the stemmer produced many mistakes, it improved the performance of their system immensely. Table 3 lists the steps of Khoja Stemming algorithm (Khoja and Garside, 1999).

After applying all the text pre-processing steps, words of length 1 are removed and Alj-News dataset ended with the number of features (terms) shown in Table 4.

4.2. Feature Selection (FS)

Since most machine learning algorithms cannot afford working with all terms in the corpus, due to memory and processing limitations, feature selection (FS) has become a routine part of automatic TC. We used Chi Square (CHI) as a FS metric for selecting the most discriminating features in the dataset. CHI has been proved to record high accuracy in classifying both English (Eldos, 2002; Eldin, 2007; El-Halees, 2008; AL-Tahrawi and Abu Zitar, 2008; Al-Tahrawi, 2013, 2014, 2015) and Arabic (Mesleh, 2007; Thabtah et al., 2009; Al-Harbi et al., 2008; Khorsheed and Al-Thubaity, 2013; Belkebir and Guessoum, 2013; Sharef et al., 2014) texts. The CHI FS metric measures the lack of independence between a term and a class. It was originally used in the statistical analysis of independent events. Its application as a FS metric for TC purposes goes through the following steps:

Table 4 The Number of Features (terms) after applying pre-processing on Alj-News Dataset.

CLASS	Number of terms
Art	3745
Economic	2178
Politics	2984
Science	2806
Sport	3332
TOTAL	15,045
FILTERED (after removing duplicates among classes)	8218

(1) For each term in each class in the training set, compute the CHI score to measure the correlation between the term and its containing class. CHI measure is computed for each term t in each class c_i as follows (Zheng et al., 2004):

$$\chi^2(t, c_i) = \frac{N \times (AD - CB)^2}{(A + C) \times (B + D) \times (A + B) \times (C + D)} \tag{10}$$

where: N is the total number of training documents in the dataset, A is the number of documents belonging to class c_i and containing t, B is the number of documents belonging to class c_i but not containing t, C is the number of documents not belonging to class c_i but containing t and D is the number of documents neither belonging to class c_i nor containing t.

(2) Combine the class-term CHI measures for terms that appear in more than one class in one score using the maximum or average score.

5. Experiments and results

Details of Feature Reduction, Performance Evaluation Measures and Results of experiments conducted in this research are presented in Sections 5.1 through 5.4.

5.1. Feature reduction

We applied a class-based local policy for selecting the features for building the PN classifier by selecting 1% of the topmost features from each of the five classes. This policy has proved to achieve the best classification performance compared to other reduction policies, like choosing the topmost corpus features, or an equal number of features from each class, as it gives each class a representative share in the final set of features used to build the classifier (Lewis and Ringuette, 1994; AL-Tahrawi and Abu Zitar, 2008; Al-Tahrawi, 2013,

Table 5 Features used to build the PN classifier.

CLASS	1% of Features
Art	37
Economic	22
Politics	30
Science	28
Sport	33
TOTAL	150
FILTERED	135

2014, 2015). The number of features selected from each class and the total number of features used to build the PN classifier after applying CHI and Feature Reduction, then removing duplicates is summarized in Table 5.

5.2. Performance evaluation measures

PN classifier performance is evaluated by computing its *precision, recall* and $F1$-measure. *Precision* is defined as the proportion of test files classified into a class that really belong to that class, whereas Recall is the proportion of test files belonging to a class and are claimed by the classifier as belonging to that class. Precision of a class c_i (P_i) is computed as (Debole and Sebastiani, 2005):

$$P_i = \frac{TP_i}{TP_i + FP_i} \tag{11}$$

and Recall of a class c_i, (R_i) is computed as (Debole and Sebastiani, 2005):

$$R_i = \frac{TP_i}{TP_i + FN_i} \tag{12}$$

where $TP_{i,}$ FP_i and FN_i refer to Truly Positive, Falsely Positive and Falsely Negative claims of the classifier respectively.

The $F1$ measure, introduced by Van Rijsbergen (1979), is the harmonic average of both *precision* and *recall*. High $F1$ means high overall performance of the system. $F1$ is computed as follows (Debole and Sebastiani, 2005):

$$F1 = \frac{2 \times \text{recall} \times \text{precision}}{\text{recall} + \text{precision}} \tag{13}$$

$$= \frac{2TP}{2TP + FP + FN} \tag{14}$$

Individual results of classes are microaveraged and macroaveraged to give an idea of the classification performance on the dataset as a whole.

5.3. Results

Results of applying our PN classifier on Alj-News Arabic Dataset are summarized in Table 6.

5.4. Analysis of results

Results of applying PNs classification algorithm on Alj-News dataset reveal that PNs have recorded high performance accuracy, using just 1% of each class features. The top performance was on 'Sport' class with 0.967 Recall and 0.959 F-measure, while the lowest F-measure performance recorded was 0.842 on 'Economics' class. Comparisons of the results reached in this research and related research works on the same dataset are presented next.

Table 6 Results of applying PN classifier on Alj-News.

Class	Precision	Recall	F1
Art	0.923	0.80	0.857
Economic	0.889	0.80	0.842
Politics	0.773	**0.967**	0.859
Science	**0.966**	0.933	0.949
Sport	0.951	**0.967**	**0.959**
Micro average	0.893	0.893	0.893
Macro average	**0.90**	0.893	0.893

Bold values indicate best results.

(Chantar and Corne, 2011) proposed BPSO (Binary Particle Swarm Optimization)-KNN as a FS method for Arabic TC. They experimented three different classifiers on exactly the same dataset (Alj-News), with the same training and testing split, used in our research. These algorithms are: Support Vector Machines (SVM), Naïve Bayes (NB) and Decision Trees (J48). They ended up with 5329 features after applying a set of pre-processing steps on the corpus. These preprocessing steps include removing hyphens, punctuation marks, numbers, digits, non-Arabic letters and diacritics. Then stop words and rare words (words that occur less than five times in the dataset) were removed. From these terms, they selected 2967 features to build the three classifiers. Results reached in their experiments on Alj-News are summarized in Tables 7–9 and comparisons of our results with the results of this research are summarized in Figs. 1–3. As is clear from the Figures, PN classifier is a competitive algorithm to the best performers in their research.

Although (Chantar and Corne, 2011) have worked on the same dataset, we used in this research, with exactly the same training and testing split, differences in the number of features used for building classifiers, FS and weighting methods adopted, as well as in the text pre-processing steps applied on the dataset documents (refer to Table 1 for the details) make direct performance comparisons between our PN classifier and their classifiers unfair, since these differences are known to affect the classification performance to a great extent. Our intended near future work is to conduct direct comparisons between our PN classifier and other well-known Arabic text classifiers using the same TC settings.

Other research works on Alj-News Arabic Dataset used different set of classes, different number of documents or different splits for training and testing subsets. We present here a comparison of the results on the common classes in our and their research experiments.

Table 7 Accuracy by class for SVM on Alj-News Dataset in Chantar and Corne (2011).

Class	Precision	Recall	F-Measure
Art	0.934	0.95	0.942
Economic	0.962	0.85	0.903
Politics	0.789	0.933	0.855
Science	1	0.933	0.966
Sport	1	0.983	0.992
W. Avg.	0.937	0.93	0.931

Table 8 Accuracy by Class for Naïve Bayes on Alj-News Dataset in Chantar and Corne (2011).

Class	Precision	Recall	F-Measure
Art	0.86	0.717	0.782
Economic	0.852	0.867	0.86
Politics	0.662	0.85	0.745
Science	0.914	0.883	0.898
Sport	1	0.9	0.947
W. Avg.	0.858	0.843	0.846

Table 9 Accuracy by Class for J48 on Alj-News Dataset (Chantar and Corne, 2011).

Class	Precision	Recall	F-Measure
Art	0.711	0.533	0.61
Economic	0.789	0.75	0.769
Politics	0.471	0.667	0.552
Science	0.849	0.75	0.796
Sport	0.917	0.917	0.917
W. Avg.	0.747	0.723	0.729

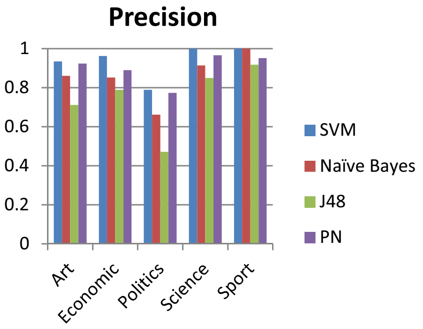

Figure 1 PN's Precision versus others on Alj-News Dataset.

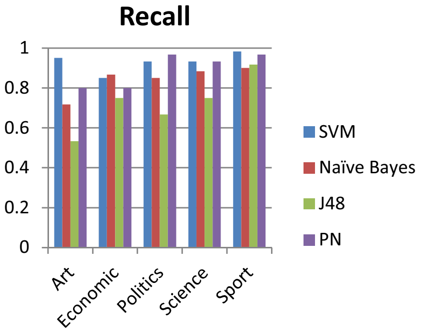

Figure 2 PN's Recall versus others on Alj-News Dataset.

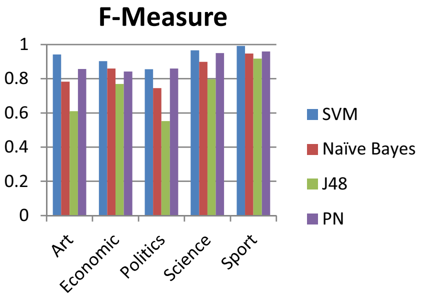

Figure 3 PN's F-measure versus others on Alj-News Dataset.

(Awad, 2012) used a version of Alj-News dataset with 16 categories, 7566 documents and 189,815 features to test 3 algorithms on Arabic TC: SVM, kNN and GIS (Generalized Instance Set). Results of their experiments are summarized in Table 10.

Table 10 Results of research work of (Awad, 2012) on Alj-News Dataset.

Algorithm	Precision	Recall	F1
SVM	0.781316	0.861111	0.819314
KNN	0.83814	0.855740	0.846849
GIS	0.845085	0.853060	0.849054

Table 11 Results of research work of (Mesleh, 2007) on Alj-News Dataset.

Category	Precision	Recall	F-measure
Economics	93.02326	71.42857	80.80808
Politics	90	76.27119	82.56881
Sports	100	85.71429	92.30769

Table 12 Overall F-Results in research work of (Mesleh, 2007) on Alj-News Dataset.

Algorithm	F-measure
SVM	88.11
NB	84.54
kNN	72.72

(Mesleh, 2007) tested CHI FS in Arabic TC using an in-house collected corpus from online Arabic newspaper archives, including Al-Jazeera, Al-Nahar, Al-Hayat, Al-Ahram, and Al-Dostor as well as a few other specialized websites. The collected corpus consists of 1445 documents. These documents fall into nine classification categories that vary in the number of documents. Data preprocessing was applied by removing digits, punctuation marks, non-Arabic letters, stop words and infrequent terms which occur less than 4 times in the training part of the corpus. In addition, Light Stemming was applied. His best results, which were achieved when extracting the top 162 terms for each classification class, are presented in Table 11 for the common classes between his and our research works. The overall performance of the three algorithms used in their research is summarized in Table 12.

It is apparent from these indirect comparisons that PNs recorded better or competitive performance using much less number of features (only 135 features compared to hundreds of thousands of features in other researches).

6. Conclusion

In this research, Polynomial neural Networks (PNs) are used, for the first time in the literature of Arabic text classification (TC), as an Arabic TC algorithm. Stemming is applied on the Alj-News Arabic dataset, Chi Square FS is used to select features and a local class-based reduction feature policy is used to select only 1% of each class features to build the PN classifier. Results achieved in this research have shown that PNs are among the top performers in classifying Arabic text documents. More importantly, PNs are able to achieve this performance in one shot (non-iteratively) and using a very small portion of the dataset features, compared to other iterative TC algorithms which need a lot of features to achieve an acceptable classification performance. Results also reveal that PNs require stemming as a necessary text pre-processing step, since PNs are usually used with a small number of features due to their high memory requirements. Nevertheless, PNs were able to record competitive results despite all the weakness points of stemming. Our intended near future work is to conduct direct comparisons between our proposed PN classifier and a set of the state-of-the art Arabic TC algorithms.

References

Ababneh, J., Almomani, O., Hadi, W., Kamel, N., El-Omari, T., Al-Ibrahim, A., 2014. Vector space models to classify Arabic text. Int. J. Comput. Trends Technol. (IJCTT) 7 (4), 219–223.

Al-Harbi, S., Almuhareb, A., Al-Thubaity, A., Khorsheed, M., Al-Rajeh, A., 2008. Automatic Arabic text classification. In: JADT;08, France, pp. 77–83.

Al-Saleem, S., 2010. Associative classification to categorize Arabic data sets. Int. J. ACM Jordan 1 (3), 118–127.

Al-Saleem, S., 2011. Automated Arabic text categorization using SVM and NB. Int. Arab J. e-Technol. 2 (2), 124–128.

Al-Shalabi, R., Kannan, G., Gharaibeh, H., 2006. Arabic text categorization using KNN algorithm. In: The Proc. of Int. Multi Conf. on Computer Science and Information Technology CSIT06.

Al-Tahrawi, M.M., 2013. The role of rare terms in enhancing the performance of polynomial networks based text categorization. J. Intell. Learn. Syst. Appl. 5, 84–89. http://dx.doi.org/10.4236/jilsa.2013.52009.

AL-Tahrawi, M.M., 2014. The significance of low frequent terms in text classification. Int. J. Intell. Syst. 29 (5), 389–406. http://dx.doi.org/10.1002/int.21643.

AL-Tahrawi, M.M., 2015. Class-based aggressive feature selection for polynomial networks text classifiers – an empirical study. U.P.B. Sci. Bull. Ser. C 77 (2), 93–110, ISSN: 2286-3540.

AL-Tahrawi, M.M., Abu Zitar, R., 2008. Polynomial networks versus other techniques in text categorization. Int. J. Pattern Recognit. Artif. Intell. (IJPRAI) 22 (2), 295–322. http://dx.doi.org/10.1142/S0218001408006247.

Assaleh, K., Al-Rousan, M., 2005. A new method for Arabic sign language recognition. In: EURASIP J Appl Signal Processing. Hindawi Publishing Corporation, New York, pp. 2136–2145.

Awad, W.A., 2012. Machine learning algorithms in web page classification. Int. J. Comput. Sci. Inf. Technol. (IJCSIT) 4 (5), 93–101. http://dx.doi.org/10.5121/ijcsit.2012.4508.

Belkebir, R., Guessoum, A., 2013. A hybrid BSO-Chi2-SVM approach to Arabic text categorization. In: IEEE Computer Systems and Applications (AICCSA), 2013 ACS International Conference, 27–30 May 2013, Ifrane, pp. 1–7. doi:http://dx.doi.org/10.1109/AICCSA.2013.6616437.

Campbell, W.M., Assaleh, K.T., Broun, C.C., 2001. A novel algorithm for training polynomial networks. In: Int NAISO Symp Information Science Innovations ISI'2001, Dubai, UAE, March 2001. doi:http://dx.doi.org/10.1.1.28.5119.

Chantar, H.K., Corne, D.W., 2011. Feature subset selection for Arabic document categorization using BPSO-KNN. IEEE 546–551. http://dx.doi.org/10.1109/NaBIC.2011.6089647.

Crammer, K., Singer, Y., 2003. A family of additive online algorithms for category ranking. JMLR 3, 1025–1058.

Debole, F., Sebastiani, F., 2005. An analysis of the relative hardness of Reuters-21578 subsets. JASIS 56 (6), 584–596. http://dx.doi.org/10.1002/asi.20147.

Duwairi, R., 2005. A distance-based classifier for Arabic text categorization. In: The Proc. of the Int. Conf. on Data Mining DMIN'05, June, Las Vegas, USA, pp. 20–23.

Duwairi, R., 2007. Arabic text categorization. Int. Arab J. Inf. Technol. 4 (2), 125–131. http://dx.doi.org/10.1002/asi.20360.

Eldin, S., 2007. Development of a computer-based Arabic Lexicon. In: The Int. Symposium on Computers & Arabic Language, ISCAL, Riyadh, KSA.

Eldos, M., 2002. Arabic Text Data Mining: A Root Extractor for Dimensionality Reduction. ACTA Press, A Scientific and Technical Publishing Company.

El-Halees, A.M., 2007. Arabic text classification using maximum entropy. Islamic Univ. J. 15 (1), 157–167, doi:http://dx.doi.org/10.1.1.124.361.

El-Halees, A.M., 2008. A comparative study on arabic text classification. Egypt. Comput. Sci. J. 30 (2).

El-Kourdi, M., Bensaid, A., Rachidi, T., 2004. Automatic Arabic document categorization based on the Naïve Bayes algorithm. In: The 20th Int. Conf. on Computational Linguistics, Geneva, August, 27, 2004.

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

Fodil, L., Sayoud, H., Ouamour, S., 2014. Theme classification of Arabic text: a statistical approach. In: Terminology and Knowledge Engineering 2014, Berlin, Germany, pp. 77–86.

Fukunaga, K., 1990. Introduction to Statistical Pattern Recognition. Academic Press.

Ghwanmeh, S., 2007. Applying clustering of hierarchical K-means-like algorithm on arabic language. Int. J. Info. Technol. 3 (3), 168–172.

Harrag, F., El-Qawasmeh, E., 2009a. Neural Network for Arabic Text Classification. In: The Second International Conference on the Applications of Digital Information, London, UK, pp. 805–810.

Harrag, F., El-Qawasmeh, E., Pichappan, P., 2009. Improving Arabic text categorization using decision trees. IEEE, NDT'09, 110–115. http://dx.doi.org/10.1109/NDT.2009.5272214.

< http://www.InternetWorldStats.com >.

< http://zeus.cs.pacificu.edu/shereen/ArabicStemmerCode.zip >.(January, 2014).

Joachims, T., 2002. Learning to Classify Text Using SVM. Kluwer Academic Publishers.

Kanaan, G., Al-Shalabi, R., Ghwanmeh, S., 2009. A comparison of text-classification techniques applied to Arabic text. J. Am. Soc. Inform. Sci. Technol. 60 (9), 1836–1844. http://dx.doi.org/10.1002/asi.v60:9.

Khoja, S., Garside, R., 1999. Stemming Arabic text. Computing Department, Lancaster University, Lancaster. < http://www.comp.lancs.ac.uk/computing/users/khoja/stemmer.ps >. (January, 2014).

Khorsheed, M., Al-Thubaity, A., 2013. Comparative evaluation of text classification techniques using a large diverse Arabic dataset. Lang Resour. Eval. Springer 47 (2), 513–538. http://dx.doi.org/10.1007/s10579-013-9221-8.

Khreisat, L., 2006. Arabic Text Classification Using N-Gram Frequency Statistics: A Comparative Study. In: Proceedings of the 2006 International Conference on Data Mining (DMIN 2006), June 26–29, Las Vegas, Nevada, USA, pp. 78–82.

Larkey, L., Connell, M.E., 2001. Arabic information retrieval at UMass in TREC-10. In: Proceedings of TREC. NIST, Gaithersburg, doi:http://dx.doi.org/10.1.1.14.9079.

Lewis, D.D., Ringuette, M., 1994. A comparison of two learning algorithms for text categorization. In: Proc Third Ann Symp Document Analysis and Information Retrieval (SDAIR'94), Las Vegas, USA, pp. 81–93. doi:http://dx.doi.org/10.1.1.49.860.

Lewis, D., Yang, Y., Rose, T.G., Li, F., 2004. A new benchmark collection for text categorization research. JMLR 5, 361–397.

Liu C.L., 2006. Polynomial Network Classifier with Discriminative Feature Extraction, Joint IAPR International Workshops, SSPR 2006 and SPR 2006, Hong-Kong. doi:http://dx.doi.org/10.1007/11815921_80.

Mesleh, A.A., 2007. Chi square feature extraction based Svms Arabic language text categorization system. J. Comput. Sci. 3 (6), 430–435.

Mohamed, S., Ata, W., Darwish, N., 2005. A new technique for automatic text categorization for Arabic documents. In: Proc. of the 5th IBIMA International Conference on Internet and Information Technology in Modern Organizations, Cairo, Egypt, pp. 13–15.

Said, D., Wanas, N., Darwish, N., Hegazy, N., 2009. A Study of Arabic Text preprocessing methods for Text Categorization. In: The 2nd Int. conf. on Arabic Language Resources and Tools, April, 22–23, Cairo, Egypt, pp. 230–236.

Sakhr Software Company's website: <www.sakhrsoft.com>, 2004.

Sawaf, H., Zaplo, J., Ney, H., 2001. Statistical classification methods for Arabic news articles. Arabic Natural Language Processing Workshop, ACL'2001, Toulouse, France, pp. 127–132.

Sawalha, M., Atwell, E., 2008. Comparative evaluation of Arabic language morphological analyzers and stemmers. In: The Proc. of COLING'2008 22nd Int. Conf. on Computational Linguistics, (poster volume), pp. 107–110.

Sebastiani, F., 2002. Machine learning in automated text categorization. ACM Comput. Surv. 34 (1), 1–47. http://dx.doi.org/10.1145/505282.505283.

Sharef, B., Omar, N., Sharef, Z., 2014. An automated Arabic text categorization based on the frequency ratio accumulation. Int. Arab J. Info. Technol. 11 (2), 213–221.

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, pp. 930–935. doi:http://dx.doi.org/10.1.1.411.3605.

Van Rijsbergen, C.J., 1979. Information Retrieval, second ed. Butterworths, London.

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

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

Zheng, Z., Wu, X., Srihari, R., 2004. Feature selection for text categorization on imbalanced data. SIGKDD Explorations 6 (1), 80–89. ACM, New York, NY, USA. doi:http://dx.doi.org/10.1.1.103.5069.